

Zusammenfassung Interaction

Philipp Jäcks

26. Februar 2016

Inhaltsverzeichnis

1	Interaction Engineering - Fragenkatalog	2
1.1	Introduction	2
1.2	Basics of Graphics and Interaction Programming	5
1.3	Fundamental Interaction Concepts	9
1.4	Advanced Interaction Concepts	13
2	Graphische Benutzungsoberflächen - Kontrollfragen	17
2.1	Begriffe und Geschichte	17
2.2	Modelle der Human-Computer-Interaction	18
2.3	Wahrnehmung	22
2.4	Gedächtnis und Handlungsprozesse	25
2.5	Normen und Bildschirmarbeit	29
2.6	Fenstersysteme	34
2.7	Dialogtechniken	37
2.8	Metaphern	41

1 Interaction Engineering - Fragenkatalog

1.1 Introduction

1. Erkläre Stärken und Schwächen des Menschen und des Computers in der Human-Computer Interaction!

Mensch	Computer
<ul style="list-style-type: none">• Kreativität• Abstrahieren und Erzeugen von Modellen• auf Unerwartetes reagieren	<ul style="list-style-type: none">• 'exakte'¹ Berechnungen• stundenlanges Ausführen derselben Aufgabe ohne zu Ermüden• exakter Speicher (Gedächtnis)

Tabelle 1: Stärken des Menschen und Computers in HCI. (Stärken = Schwächen des anderen!)

2. Kernunterschied bei der Entwicklung von computational solutions und interactive solutions
 - Interaction: Kernfaktor ist das Human Computer Interface
 - Computational: Kern ist effizienter Algorithmus(?)
 - Rechenleistung steigt immer weiter während menschliche Aufnahmefähigkeit stagniert/konstant ist
 - Kernfaktor bei der Informationsverarbeitung durch den Menschen ist das Interface
 - \Rightarrow Interaction soll smooth und effizient; Feedback soll reich an Informationen und instantan sein
 - HC-Interaction - Mensch und Computer gehen Hand in Hand, jeder erfüllt die Aufgaben, die er am besten lösen kann (siehe Tabelle 1)

Computation - closed system	Interaction - open system
<ul style="list-style-type: none">• Eingabe• Verarbeitung• Ausgabe• deterministisch, Endzustand	<ul style="list-style-type: none">• Veränderung in der Umwelt• Empfange Events• Reagiere auf Events• endlos, nichtdeterministisch

3. Action Cycle by Norman
 - a) Mensch hat Ziel im Kopf (Goal)
 - b) Planen der notwendigen Schritte (Plan)
 - c) Spezifizieren der Schritte (Specify)

¹Denke an max. Genauigkeit von Fließkommazahlen!

- d) Umsetzen der Schritte in der Welt (Perform)
- e) Feedback in der Welt beobachten (Perceive)
- f) Feedback interpretieren (Interpret)
- g) Ergebnis mit Zielen vergleichen (Compare)
- h) Beginne bei Schritt 1 bzw. 2

ACTION CYCLE

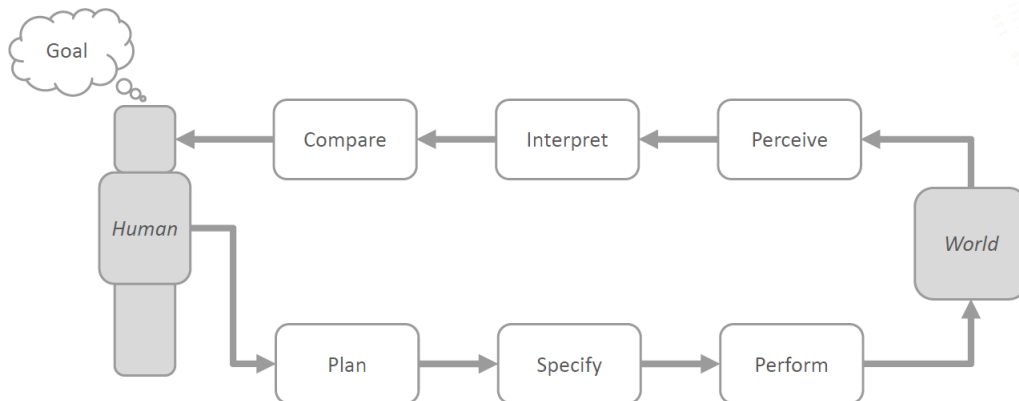


Abbildung 1: Action Cycle nach Norman

4. Iteration/Bsp für den Action Cycle

Am Beispiel: Kaffee holen in der Mensa

- **Goal:** Kaffee in der Mensa holen
- **Plan:** Aus dem Büro gehen
- **Specify:** Operation - Türgriff betätigen um Bürotür zu öffnen
- **Perform:** Türgriff drücken
- **Perceive:** Griff öffnet das Schloss, Tür öffnet sich
- **Interpret:** Tür ist offen
- **Compare:** Schritt erfolgreich, Führe weitere Schritte aus

5. Gulf of Execution and Evaluation

• Execution

beschreibt die Mühe/Aufwand der angestrebten Aufgaben

Kann ich das tun? Wo ist die notwendige Funktionalität? Welches Gerät nutze ich? Wie führe ich das Kommando aus?

- **Evaluation**

Beschreibt die Mühe/Aufwand die Veränderung der Umwelt zu interpretieren
Ist überhaupt etwas passiert? Wo ist etwas passiert? Was ist passiert? Passen Effekt und Absicht zusammen?

- Interaction cost = Summe des physischen und mentalen Aufwandes um ein Ziel zu erreichen
- Beispielhaft am Action Cycle:
 - Cost of Decision (Goal): Fokus muss auf Teilmenge von Informationen und Interfaces gelenkt werden
 - Cost of System Power (Plan): Übersetzen von Zielen im Kopf in Operationsequenzen ist schwer, insbesondere für komplexe Systeme
 - Cost of visual clutter/visuelle Überfüllung/reizung (Perceive): Bsp - Mouse Hover Effekte erzeugen Überreizung und erschweren Zustandswahrnehmung

6. The Three levels of interaction

- **low level:** Selection and Manipulation
- **inter-mediate level:** Exploration and Navigation
- **high level:** Problem-solving

7. The levels of (human) interaction processing

- Instinktiv (Perform and Perceive): vollkommen unterbewusst, ohne Kontrolle, schnell, Basisfähigkeiten - Bsp: Arm bewegen um Türgriff zu fassen
- Behavioral (Specify and Interpret): teilweise unterbewusst, leichte Kontrolle, schnell, gelernte Fähigkeiten - Bsp: Drücken der Klinke öffnet Tür
- Reflective (Plan and Compare): volles Bewusstsein, langsam, komplexe Analyse - Bsp: Tür ist offen, was bedeutet das?

8. at least 5 golden rules or guidelines for interaction

Golden Rules - Norman

- Discoverability: Welche (möglichen) Aktionen können bestimmt werden?
- Feedback: reichhaltiger und kontinuierlicher Fluss an Informationen über den Zustand
- Affordances: angemessene Aufforderungen um die gewünschte Aktion durchzuführen
- Signifiers: effiziente Signalgeber für Discoverability und Feedback
- Mappings: gute Zuordnung zwischen Controls and Actions

Guidelines - Shneiderman

- Konsistenz: ähnliche Situationen sollen ähnliche Aktionen erfordern

- b) Universal Usability: Assistenz anbieten (Hilfe, Shortcuts,...)
- c) Informative Feedback: Feedback für jede Useraktion
- d) Closure: klarer Beginn, Ablauf und Ende einer Aktion; kombiniere mit Punkt 3
- e) Prevent Error: vermeide fehlerhaften input, Recover from user error
- f) Easy reversal of actions: zb undo redo
- g) internal locus of control: User kontrolliert das System
- h) reduce short-term memory load: keep it simple

Sonstige Notizen

- Vor- und Nachteile der Interaction

Vorteile	Nachteile
<ul style="list-style-type: none"> • ist mächtiger als "Algorithmen" • anspruchsvolleres Verhalten 	<ul style="list-style-type: none"> • User muss wissen <i>was</i> er/sie möchte • User muss wissen <i>wie</i> er/sie den Computer bedienen muss um das Ziel zu erreichen <ul style="list-style-type: none"> • Anwendung ist zustandsbehaftet -> User kann sich verlieren/steckenbleiben (stateful things can be broken)

- Bottlenecks - Processing: CPU, RAM, Netzwerk etc.. mittlerweile in vielen Anwendungsgebieten nicht mehr so relevant
- Bottlenecks - Information: enorm wichtig welche Daten auf dem kleinen Bildschirm am Ende angezeigt werden (viele, viele Daten gespeichert; welche Davon sind wichtig und werden angezeigt?)
- Bottlenecks - Aufnahmefähigkeit Mensch

1.2 Basics of Graphics and Interaction Programming

1. Illustrate the interplay of the action cycle and the model-view-controller pattern!
2. Discuss different forms of presentation!
Wahrnehmungen durch die 5 Sinne, geordnet nach Relevanz für HCI:
Visuell, Audio, Fühlbar, Geruch, Geschmack
wichtige Aspekte: Bandbreite, Aufmerksamkeit, Vergänglichkeit/Flüchtigkeit
3. Discuss different forms of expression!
Sprache, Point & Gestures, Physical movement (sich selbst, Objekte wie Maus, Tastatur..)
wichtige Aspekte: Genauigkeit, Geschwindigkeit, Aufwand

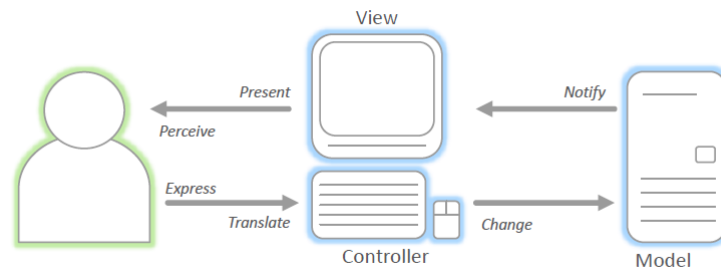


Abbildung 2: Zusammenspiel Action Cycle und MVC

4. Discuss pros and cons of uni-modal and multi-modal interaction!

Uni-Modal: genau eine Form der Presentation und Expression (zb Visuell und Point & Gestures) \Rightarrow ein Kanal

Multi-Modal: mehrere Formen/Kanäle; zb Visuell, Audio und Touch, Sprache, Point..

	Vorteile	Nachteile
Uni-Modal	einfache Implementierung	nur ein Kanal
Multi-Modal	umfangreiche Formen der Interaktion	komplexe Implementierung und schwieriger zu lernen (für den User)

5. What are the mental, implementation and represented model? Why are they important?

- **Mental:** gedankliche Vorstellung des Modells; entspricht menschlicher Natur; beschreibt v.a. Operationen zum Erreichen des Ziels
- **Implementation:** interne Darstellung; technisch limitiert und vom Entwickler vorgegeben; enthält Daten, Parameter, Algorithmen,...
- **Represented:** Darstellung des (Implementation) Models auf dem PC; vom Entwickler vorgegeben;

Important, weil Mental Model nicht 1:1 in Computer dargestellt werden kann; müssen abstrahieren

6. What is a graphics context and what can it be used for?

Wird gestellt vom Window System und ist verknüpft mit dem Zeichenareal. Stellt Funktionalitäten zum Zeichnen bereit.

7. Explain the basic procedure for drawing paths!

begin path; move to; add (line, curve, arc,...); end path

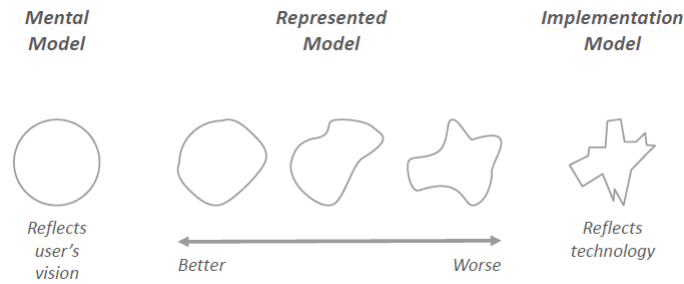


Abbildung 3: Zusammenspiel der 3 Modelle

8. Name three interaction devices and characterize the input they deliver
 Pointing Device (Maus): absolute oder relative Koordinaten
 Triggers (Tasten)
 Value Input (Sensoren)
9. Give examples of atomic and composite inputs!
 atomic: single click, Bewegung des Zeigers, aktiviere Trigger,...
 composite: double click, drag n drop, Gesten,...
10. Illustrate the interconnections between model, view and controller in MVC pattern!

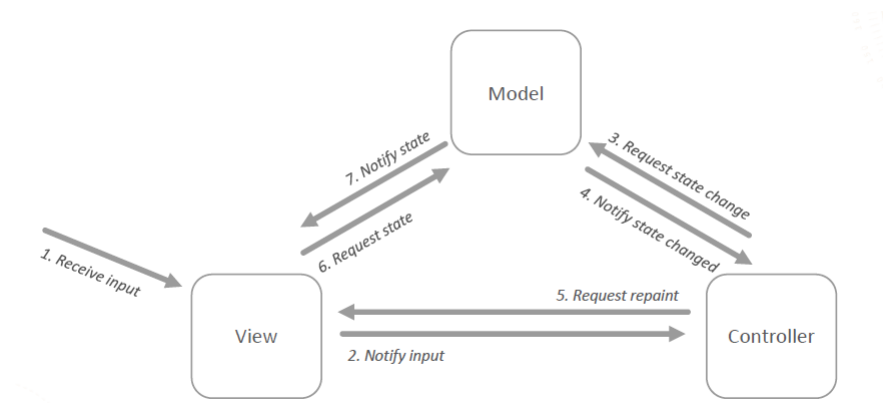


Abbildung 4: Interconnection der drei Bestandteile des MVC Pattern

11. Discuss advantages and disadvantages of the MVC!

Vorteile	Nachteile
klare konzeptuelle Trennung der Bestandteile	<ul style="list-style-type: none"> enge Kopplung der Komponenten (durch Interaktion)
<ul style="list-style-type: none"> Entwurf für generelle Architektur 	<ul style="list-style-type: none"> komplex zu implementieren

12. What loop is the fundamental ingredient of interactive systems!
event loop: Endlosschleife von Reaktionen auf Events; verschiedene Eventtypen; verschiedene Reaktionen
13. Name at least five types of events in interaction systems!
- Application level: loaded, finished,...
 - Widget level: Repaint, Resize, Activated, focused,...
 - Input level: key pressed/released/typed,.. Touch start/moved/end, Mouse pressed/...
 - Window System signals external event: repaint of window required; resize of window; Input von Peripherie (zb Maus)
 - Application signals internal events: state change, timer elapsed
14. How are events propagated in interactive systems?
 Werden in EventQueue reingesteckte und nach FIFO verarbeitet;
 Dispatch Events: Traversiere Window Tree und wähle vorderstes Fensters aus
 Receive Events: definiere Verhalten bei Event; ausgeführt durch Listener
15. How can callback functions be used to react to events?
 Callback als Parameter einer Funktion -> Funktion ausgeführt, dann führe Callback aus. Auf Events: Ich sage Programm mach etwas und du sollst mir Bescheid geben(=callback), Ausführung erzeugt Event und callback wird gerufen(omfg.. xD)
 gut für basic (device) events; but limited flexibility
16. What are delegates and listeners? Interface um auf Events zu reagieren; Logik muss vom Entwickler umgesetzt werden
 cover many different events, also user-generated ones; but sometimes complex
17. What are signals and slots and which toolkits make use of them? QT, GTK+,...
 Kommunikation zwischen Objekten; Signal wird emitted wenn Event auftritt; Slot = Funktion die bei bestimmtem Event aufgerufen wird
 ähnlich zu Callbacks, aber mit höherer Flexibilität
18. Express in pseudo code the main event loop and explain its components!
 Block and render -> wartet auf Event und rendert nur bei Bedarf neu

```

while (true || event != QUIT) {
    event = wait_for_event();
    do_repaint = update_model(event)
    if (do_repaint) {
        render_model()
    }
}

```

check and render full-throttle -> render mit jedem Durchlauf und reagiert nur auf Events


```

while (true || event != QUIT) {
    if (check_for_event()) {
        event = get_event()
        update_model(event)
    }
    render_model()
}

```

19. What is a disadvantage of method overriding for reacting to events? sollte in Sub-Class ausgelagert werden um Custom Verhalten umzusetzen; im Gegensatz zum Listener Konzept wesentlich unübersichtlicherer Code

1.3 Fundamental Interaction Concepts

1. At which three levels can interaction be considered?
 - Low level: basic picking and manipulation
 Event Notifications: *Etwas* ist passiert;
 Event Type: *Was* ist passiert;
 Interaction abhängig vom räumlichen Kontext: also *wo* etwas passiert ist;
 Interaction Handling = Event Notification + Event Type + räumlicher Kontext
 - Intermediate level: Kombiniere low level Techniken zu navigieren, exploration, modellieren
 - High level: kombiniere intermediate level Techniken zu komplexen Prozessen (Understanding, Kreativität)
2. Why is picking necessary for human-computer interaction?
 Picking - Grundelement für Interaktion, Mensch ist es in realer Umgebung gewohnt Dinge anzufassen bevor er mit ihnen interagiert
 Für Picking extra Ebene -> MDPC - model, display view, picking view, controller
 Picking view beschreibt Interaction Geometry (ermöglicht räumlichen Kontext) -> Regionen mit denen interagiert werden kann
 Vorteil der View Trennung: einfacheres Modellieren, Testen; GUI kann ausgetauscht werden ohne Interaktion zu verlieren; verbessertes Verhalten (Bsp: Dropdown Mouse Movement)
3. Characterize the picking problem! What is given, what is sought(gesucht)? Given: Interaction Geometrie $G = \{g_1, \dots, g_n\}$ und eine Position auf dem Bildschirm $P = (x, y)$
 Gesucht: Interaction Geometrie $G' \subset G$ am Punkt P , wobei
 $|G| = 0$, wenn nichts matcht
 $|G| = 1$, für einen eindeutigen Match
 $|G| > 1$, für mehrere Matches
Technical Requirements: Speed -> Picking muss innerhalb ms identifiziert werden;

Picking requests können sehr schnell eintreffen

Genauigkeit: muss exakt sein, front-most object

Human R: Fitt's law: betrachte beim Design die Zeit, die Nutzer für Interaktion vor. benötigen wird

4. Explain the role of essential geometry and "MDPC"! MPDC - siehe 2 Fragen vorher
Essential Geometry - Grundlegende Geometrie (Punkte, Linien, Shapes, Objects, Rays):

Controller muss drag region auf model mappen. Regions setzen sich zusammen aus den essential Geometry Bestandteilen -> Essential Geometry dient quasi als Connection zw. View und Controller beim Picking

Vier muss Regions kennen (zwecks rendern)

5. Sketch the basic implementation strategies for picking!

Screen Space Picking

- Render Szene in Picking Buffer, Geometrie hat dort eine ID
- Benutze Pointer Koordinaten als direkter Index für den Picking Buffer
- Javascript: Draw Path -> apply HitRegion -> Event: Abfrage welche HitRegion getroffen wurde
- OpenGL: zusätzliches RenderTarget -> Elementen IDs zuweisen -> Geometrie mit Hilfe der IDs (in Form von Farben) rendern -> look up ID from picking buffer

Object Space Picking

- Projiziere Pointer in die Szene mit Hilfe inverser Transformationen
 - Berechne Containment/Intersection mit der Szenen Geometrie
 - Nearest Neighbor Search:
 - Finde Geometrie am nächsten zu P
 - Optional: beschränke Picking auf Epsilonumgebung
 - machbar für Punkte(0D) und Linien(1D)
 - Punkte: Berechne Distanz von P zu allen Punkten; nimm Punkt mit geringster Distanz
 - Linien: Lot von P auf Linien fällen -> nimm Linie mit kürzestem Lot
-
- Containment Search:
 - Finde Geometrie, die P enthält
 - machbar für Shapes (2D) und Objekte (3D)
 - Shapes: Check if P is contained in geometry

- Objects: Wandle P in Ray/Strahl um und prüfe ob Strahl die Dreiecke des Objekts schneidet
6. Discuss different ideas to accelerate the picking! **Computation Performance**
- Reduziere Komplexität der Geometrieberechnung
 - Sub-linear Search -> benutze hierarchische Strukturen (Bäume,...) um Suchen zu beschleunigen
 - Pointer Kohärenz: gucke zunächst in Nachbarschaft

User Performance

- unterschiedliche Picking Mechanismen mit teils mehr Features
 - Sticky Cursor
 - Bubble Cursor
 - Bubble Lens
7. Selektionsarten:
- Click Selection: Nutzer klickt Elemente einzeln an -> Serie von einzelnen Pickingaktionen; typisch: Simple Click, Strg+ Click, Shift + Click
 - Rectangle Selection: Nutzer zeichnet Rechteck. Selektion entweder *inside*: shape was auch immer muss zu 100% im Rechteck enthalten sein gut für kleine, reguläre Elemente -> Bsp Desktop Icons
Nachteil: bei unregelmäßigen oder großen Strukturen muss ich gut abschätzen können, ob mein Rechteck alle Elemente einkapselt *overlap*: Elemente müssen Rechteck nur schneiden
Vorteil: gut für unregelmäßige und große Elemente (zb Landkarte)
Nachteil: Frage: wie viel muss geschnitten werden?!
 - Lasso Selection: Selektion innerhalb free-form shape -> Nutzer zeichnet mit Maus quasi eine freie Form und innerhalb dieser wird selektiert
8. Discuss pros and cons of inside and overlap selection! siehe Frage vorher
9. Grundlegend: Manipulation
Object der Szene -> Interaktion benutzt um Geometrie und Eigenschaften des Objektes zu ändern
View auf die Szene -> Kameramanipulation -> verändert die Perspektive auf die Szene
- Constrained Manipulation: Manipulation nur in bestimmten Bedingungen mgl, Bspw. Scrollbar nur in bestimmte Richtungen manipulierbar
10. What is the difference between absolute and relative manipulation? Mappen des räumlichen Kontexts auf die Manipulation, entweder
absolute: Punkt im Raum wählen -> dann Ergebnis -> abhängig von der aktuellen

Pointer Position

relativ: Pfad durch den Raum wählen -> dann Ergebnis -> Ergebnis abhängig von Serie von Pointer Positionen

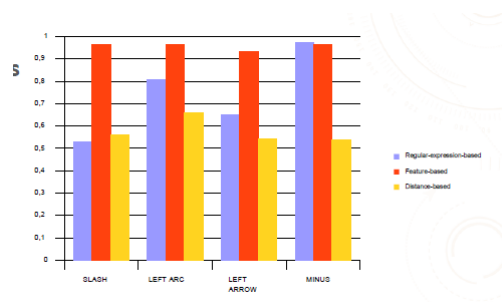
11. Wofür Navigation? -> große Mengen an Daten können nicht alle auf dem Bildschirm dargestellt werden -> Nutzer muss durch diese navigieren können
Wie? Besuche verschiedene Teile der Informationen auf verschiedenen Detaillevel
Requirements: Wo bin ich? Wohin kann ich gehen? Wie komm ich dahin? Was liegt dahinter? Wo ist es sinnvoll hinzugehen? Wo war ich?
12. What is the mental model for actor-centric vs. object-centric manipulation? Discuss consequences for interaction!
Actor: Ich hab das Auge in der Hand und navigiere mich selbst durch die fixe Szene
Object: Ich habe die Welt in der Hand und navigiere indem ich die Szene selbst bewege.
Konsequenzen: Einfluss wie Interaktion mit Hardware umgesetzt wird; Bsp Zoom mittels Mauseisrad -> roll ich es weg (Object Centered) oder zu mir (Actor Centered)?
13. In which respect is scrolling limited, what is the advantage of scrolling?
lineare Navigation; festgelegte Richtungen; relative Positionierung der View mittels Drag n Drop;
Absolute Positionierung der View mittels Scroll handle;
Feedback der View Position innerhalb der Welt
Pro: Easy to use; Con: Limitierung bei komplexen Problemen durch Linearität
14. Illustrate a space-scale diagram!
 - Grundlegende Idee: durch Skalierung die Einschränkungen des Scrollings umgehen
 - dazu: unterstützt Scaling unterschiedliche Geschwindigkeiten der Navigation -> langsam und präzise, wenn kleine Skalierung; schnell und grob, bei großer Skalierung
 - -> Zoomable UIs = Scaling + Scrolling
 - Space-Scale diagram dient zum Verstehen der ZUIs
 - Zeigt unterschiedlichen Zoomstufen abhängig von Skalierung (Achtung: kann auch unendlich sein)
15. Which questions must a user be able to answer during view navigation?
 - Where can I go?
 - How do I get there?
 - Where am I?
 - What lies beyond?
 - Where can I usefully go?

- Where have I been?
16. Give examples how to support "Where can I go?"
View Space Navigation: Verändere Viewport Position und Größe -> Bsp: smoothes und effizientes Zoomen und Schwenken
Data Space Navigation: exploite Struktur im Datenraum -> Bsp: Kantenbasiertes Reisen
 17. Give examples how to support "How do I get there?"
Schwenken: Drag n Drop (relativ); Pan Wheel (relativ); Scrollbar (absolut)
Zoom: Mouse Wheel rotation (relativ); Zoom Slider (absolute); Elastic Rectangle (Absolute; zoom in only)
 18. Give examples how to support "Where am I?"
Übersichtsfenster (Position); Scrollbar Indikatoren (Position); Zoom Slider (Skalierung); Infinite Grid (Skalierung)
 19. Give examples how to support "What lies beyond?"
Indikatoren, die anzeigen das Elemente außerhalb des Viewport liegens; Bsp: Pfeil, Proxy (Stellvertreter am Rand des Viewport); Halo (Halbkreis mit Objekt als Mittelpunkt); Wedge
 20. Give examples how to support "Where can I usefully go?"
Führung: Fokus = aktuelle View; Context = Nachbarn des Fokus -> bestimmte und bewerte Kandidaten -> Stelle Empfehlungen dar
 21. Give examples how to support "Where have I been?"
History Management wie redo/undo; speichere Zustand des Systems -> Problem: Wann speichern wir diesen? Bei Event? Zeitbasiert? -> Probleme: State change kann sehr schnell passieren (dragging a slider bspw)

1.4 Advanced Interaction Concepts

- Give examples of gestures?
 - **Basic:** Bedeutung des Inputs -> Stroke, Touch, Hand, Body, Facial Gestures
Typen: offline -> Geste performed -> erkannt -> interaction getriggert -> Ende
online: während Geste performed wird -> kontinuierliche Erkennung -> kontinuierliche Interaktion -> Ende
 - **Recognition of Gestures:**
 - Basisprozedur für offline Gesten:
 - Input -> Setze Menge von Referenzgesten und Menge von Interaction Events
 - Processing: Bestimme Referenzgeste passend zum User Input
 - Output: Geste, die User Input matcht

- *Recognition of Stroke gestures:*
 - * Stroke aufnehmen -> Free: ohne Bedingungen; Isochronal: Punkte des Strokes werden in konstanten Intervallen aufgenommen; Equidistant: Punkte des Strokes werden in konstanten Abständen aufgenommen
 - * Distance Based Approach: Menge von Referenzstrokes -> Mappen der Kontrollpunkte -> Zentriere Bounding Boxes des Strokes und Ref-Strokes -> Berechne durchschnittliche Abweichung der Kontrollpunkte
- *Feature Based*
 - * Extract feature Vector aus dem Stroke
 - * Berechne Gemeinsamkeit mit Ref-Stroke
- *RegEx based*
 - * Elemente des Strokes werden auf Chars gemappt
 - * Abgleich des Strings mit RegEx(=Ref-Stroke)
- Are stroke and touch gestures typically online or offline gestures?
stroke offline -> User gibt Input -> Referenz erkennen -> Output
Touch sind online
- Illustrate the procedure of regular-expression-based gesture recognition!
Def. Ref-Stroke als RegEx -> zeichne Stroke als String auf -> RegEx matching
- Which recognition approach you are aware of has the best recognition rate? Why is this so?



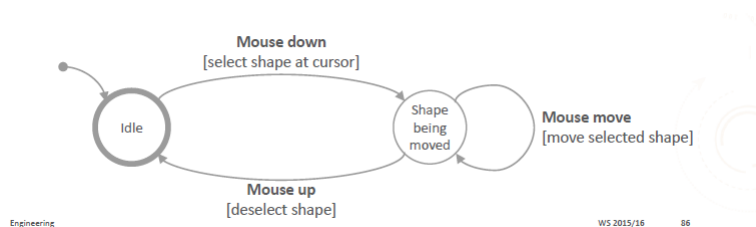
Feature Based am genauesten, weil... ich glaub, dort die Referenzstrokes mit lernen, also während der Benutzung an den Nutzer angepasst werden

- Which touch events exist in typical toolkits? What information do they contain?
Touch Start, Move, End, Cancel
Enthalten Liste mit allen Touch Punkten seit die Geste gestartet wurde
- How can a multi-touch pinch gesture be detected on Android devices?
ScaleGestureDetector mit Custom Listener -> füge Detector an der View hinzu -> TouchEvent: verschicke Event an Detector -> Detector ruft listener, einmalig, wenn Geste identifiziert

- Discuss pros and cons of animated visual feedback!
 Änderung startet instantan -> fließender Übergang vom alten zum neuen Zustand durch Animation
 Pros: hilft die mentale Map aktuell zu halten
 Cons: benötigt Zeit, change blindness -> visuelle Veränderung vom Betrachter nicht wahrgenommen
- Sketch the main loop for animated feedback!
 1. Process events
 2. Update Model
 3. render model
 4. Emit repaint Event, wenn mehr Updates notwendig sind
- Are there any studies regarding advantages and disadvantages of animation? ja
- What is the difference between timed animations and converging animations? Animation für eine konstante Zeitperiode; verschiedene Interpolationen (linear, slow-in, slow-out,...); einfacher Weg
 Converging: animiere bis Konvergenz erreicht ist; verschiedene Wege der Zustandsübergangs (spring-mass, smooth and effizient zooming)
- What are easing functions good for?
 vermeiden das unnatürliche Verhalten einer linearen Animation (instantaner Start und Stopp)
- Explain animation via spring-mass systems (Masse Feder System)!
 Konvergierende Animation -> Zustand = Punktmasse mit Pos. p , Beschleunigung a , und Geschwindigkeit v
 Übergang: Unterschied zw. Zuständen erzeugt Kraft; Kraft abhängig von der a der Punkte Masse in Richtung des neuen Zustandes -> a führt zu Bewegung
 Beende Animation, wenn Energie klein genug (und nicht am Ende des Weges); beachte: Feder schwingt ja quasi hin und her
- How can view navigation in ZUIs be enhanced with animation?
 smooth and efficient viewport animation (konvergierendes Verfahren)
 Idee: zoome heraus -> navigiere zum Punkt um zoome währenddessen wieder herein; mit Hilfe von Space Scale Diagram
 effizient = kürzester Pfad; smooth = keine wahrnehmbaren Unterbrechungen
- Contrast interactive lenses against regular interaction, discuss pros and cons?
 Lenses: leichtgewichtig; fokussieren temporäre Effekte
 regulär: schwergewichtig; haben globalen permanenten Effekt
- Give a definition of interactive lenses!
 Lens = Selection + alternative Representation + zusätzliches visuelles Feedback

- Sketch the conceptual model of lenses and the involved components!
 Input data: Selektion von Pixeln, Geometrien oder zugrundeliegenden Daten
 Data processing: Lens Funktion -> verarbeite Input für Linseneffekt; je nach Selektion nur Teilmenge der Verarbeitungsschritte notwendig; kleine Selektion => advanced Operations mgl

 -> 3 Linseneffekte: Veränderung; Ausblendung und Anreicherung Result output:
 Join -> führe Linseneffekt mit Hintergrund zusammen, auf dem Level von Pixeln, Geometrien oder underlying data
- How can lenses be interacted with?
 - Positionierung mittels Maus und Parameterveränderung durch GUI (klassisch)
 - Touch: bsp: zoom der Linse wie aufm Handy halt.. :D
 - fühlbare/greifbare Interaktion: Bsp: Kunststoffring auf Tabletop als Linse -> von Software erkannt und platziert an Rings Position die Linse
 - gaze-based: Interaktion mittels der Augenbewegungen
- What is interaction syntax and why is it important?
 Menge von legalen Eingabesequenzen verbunden mit den Aktionen, die als Ergebnis auf die Eingabe aktiviert werden
 Wichtig für komplexe Interaktion -> erfordert viel Planung wie die Interaktion dem Nutzer gezeigt werden kann
- How can interaction syntax be specified? Discuss benefits and problems!
 - Implizit durch Listener Logik -> führt zu Spaghetti Code
 - Event Diagramme -> Pointer Positionen als Trajektorie + annotierte Events und Zustandsdarstellung entlang der Trajektorie -> sinnvoll für einfache Aufgaben, aber limitiert
 - Interaction State Machines: sollte klar sein.. mit Hilfe von State Charts halt.. siehe SWT VL
- Sketch a basic interaction using a state machine with guards!

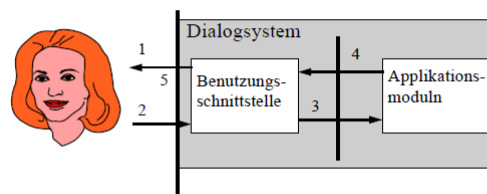


2 Graphische Benutzungsoberflächen - Kontrollfragen

2.1 Begriffe und Geschichte

1. Erläutern Sie die Begriffe Dialogsystem, UI und Usability!
 - **Dialogsystem:** auch interaktives System. Kombination von Hardware- und Softwarekomponenten, die Eingaben vom Benutzer empfangen und Ausgaben zum Benutzer übermitteln, um ihn bei der Ausführung einer Arbeitsaufgabe zu unterstützen
 - **UI - Benutzungsschnittstelle** Alle Bestandteile eines interaktiven Systems, die Informationen und Steuerelemente zur Verfügung stellen, die für den Benutzer notwendig sind, um eine bestimmte Arbeitsaufgabe mit dem interaktiven System zu erledigen.
 - **Usability** Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv (effectiveness), effizient (efficiency) und zufriedenstellend (satisfaction) zu erreichen.
2. Welche Schnittstellen hat ein Dialogsystem?
 - Benutzungsschnittstelle (UI)
 - Applikationsmodule
 - Grundprinzip: UI fragt nach Eingabe -> gibt Eingabe an Appl.Module -> Rückgabe der unformatierten Ausgabe an UI -> UI gibt formatierte Ausgabe an Nutzer

Trennung Benutzungsschnittstelle und Applikation



- 1: Prompt des Dialogsystems
- 2: Eingaben des Benutzers
- 3: Übergabe von Daten an Applikationsmodule
- 4: unformatierte Ausgaben der Ergebnisse der Applikationsmodule
- 5: formatierte Ausgaben der Ergebnisse, Fehlermitteilungen, Hilfetexte

3. Welche Modelle muss ein Dialogautor beim Entwurf von Benutzungsschnittstellen beachten?

- Benutzermodell -> mentales Modell
Aspekte des Modells: Anwendungsbereich, Arbeitsaufgabe/kontext, Kenntnisse, Erfahrungen, Fertigkeiten
 - Aufgabenmodell -> hauptsächlich HCI im Fokus
 - Architekturmodell -> UI-Software
 - Dialogspezifikationsmodell -> siehe Frage vorher(?)
4. Welche Beiträge kann die Ergonomie zur Erstellung von Benutzungsschnittstellen leisten?
- Optimieren der Arbeitsabläufe durch niedrigen Aufwand des Benutzers und hohe Funktionalität des Dialogsystems
 - ganzheitliche Gestaltung der Software
 - benutzergerecht: Berücksichtigung der Stärken und Schwächen des Menschen
 - aufgabenangemessen: Computer werden vom Benutzer zur Bearbeitung konkreter Aufgaben eingesetzt
 - technikbewusst: neue technische Optionen zum Wohle des Benutzers einsetzen
 - organisationsgerecht: Berücksichtigung der organisatorischen Einbindung
 - Computerwissenschaften -> Softwaredesign -> Dialogtechniken (Informationsdarstellung, Ablauf,...), Software Engineering (Analyse, Modellierung, Entwurf,...),
 - Arbeitswissenschaften -> Arbeitsplatzgestaltung; Arbeitspsychologie
 - Humanwissenschaften -> Physiologie
 - Psychologie -> Kognition (Gedächtnis, Verstehen, Lernen); Wahrnehmung
 - Ergebnisse der Ergonomie:
 - a) Gesetze und Verordnungen zum Arbeitsschutz
 - b) Normen ((inter-)nationale Standards)
 - c) Empfehlungen
 - d) Designregeln (Gestaltungsregeln, Style Guides, Pattern)
 - e) Tools

2.2 Modelle der Human-Computer-Interaction

Es gibt nicht ein Gesamtmodell, sondern viele Teilmodelle, die in unterschiedlichen Aspekten hilfreich sein können.

1. Was ist ein Dialogsystem? siehe subsection vorher

2. Was ist die Benutzungsschnittstelle/User Interface? siehe subsection vorher
3. Welche Schnittstellen hat ein Dialogsysteme? siehe subsec vorher
4. Welche Modelle müssen Dialogautoren beachten, Welche Rollen gibt es?
 - Modelle siehe subsection vorher
 - Gestalter und Bewerter der physischen Umgebung
 - Entwickler von UI Tools
 - Entwickler von UI /Dialogautor
 - Designer
 - Bewerter von UI
 - Applikationsanalysator
 - Applikationsprogrammierer
 - ...
 - Endbenutzer
5. Was ist das Seeheim-Modell?
 - (ähnlich MVC)
 - 3 Bestandteile:

Presentation (lexikalische Ebene): für systemnahe Ein/Ausgaben zuständig; verfügt dazu über Devices=Schnittstellen zu physikalischen oder logischen (zb Menüs) Ein/Ausgabegeräten

Dialog control (syntaktische Ebene): durch Austausch von Token Ablauf des Dialogs regeln; unabhängig von Geräte und Anwendung

Applicaition interface (semantische Ebene): Vermittlerrolle zw. Dialogcontrol und Applikation
 - Grenzen: Wie erfolgt der Kontrollfluss (siehe nächste Frage)? von welcher Art ist die Kommunikation? (Nachrichtenaustausch oder Prozeduraufruf?)
 "schnelle" grafische Ausgaben brauchen "Sonderwege" (wtf?)

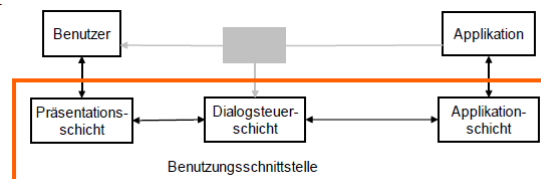


Abbildung 5: Seeheim Modell

6. Welche Vor- und Nachteile haben interne und externe Kontrollarchitekturen im Seeheim-Modell?

intern: Programm besitzt alleinige Kontrolle über UI; UI steht als Paket von Unterprogrammen zur Verfügung

extern: Anwendung in Unterprogramme unterteilt, welche von UI aufgerufen werden

	Vorteile	Nachteile
intern	<ul style="list-style-type: none"> • einfache Realisierung • optimale Anpassung an das Anwendungsprogramm <ul style="list-style-type: none"> • entspricht Vorgehensweise bei Systementwicklungen und Aufgabenzerlegungen 	<ul style="list-style-type: none"> • keine saubere Aufgabentrennung • bei Änderung an der UI Eingriffe im gesamten Programm • sehr lokale Dialoge • Einheitlichkeit und Konsistenz der UI hängen von Programmierdisziplin ab
extern	<ul style="list-style-type: none"> • saubere Separierbarkeit von UI und Anwendungsfunktionalität • benutzerspezifische UI lässt sich angemessen realisieren • anwendungsunabhängige Modifizierbarkeit • Leichte Portierbarkeit vom Prototyp zur fertigen Anwendung 	<ul style="list-style-type: none"> • Anwendungsprogramme müssen atomierbar sein • Dynamik/Interaktionsfolge liegt außerhalb der Anwendung Verteilung der Semantik • Komplexität der Schnittstellenbeschreibung <ul style="list-style-type: none"> • mögliche Redundanzen in den verschiedenen Unterprogrammen • Ausgabeaufforderung der Anwendung

Tabelle 2: Vor- und Nachteile der Kontrollarchitekturen im Seeheim-Modell

7. Nennen sie Modelle für Dialogspezifikationen!

dienen zur formalen Spezifikation des Dialogablaufs

sollen ausdrucksstark und einfach sein; typische Dialogstrukturen unterstützen (parallele Prozesse, sequentielle Schritte, Verzweigungen des Ablaufs); unterschiedliche Detailstufen

- Multi-Party-Grammatiken: kontextfreie Grammatiken; Erweiterung der BNF durch Spezifikation wer was tut bei Interaktion; einfach und natürlich, aber nicht ausdrucksstark
- Menühierarchien: baumartige Struktur zur Spezi von Alternativen; einfach, aber nicht ausdrucksstark
- Zustandsübergangsdiagramme (STN): quasi Graph der Transitionen darstellt

- verbesserte STN (=ATN)
 - State Charts: lösen einige Probleme der STN/ATN, zb Hierarchische, nebenläufige, Zustände, History
 - Ereignis Handler: Dialogbeschreibung durch Regeln
 - Petri-Netze: sollte klar sein
 - User-Action-Notation
8. Welche Benutzermodelle kennen sie?
- Ziel: konsistentes Dialogverhalten auch in breitem Anwendungsbereich anhand Wissen des Nutzers über Anwendungsbereich, seine Ziele und seine Pläne
- Einteilung nach Datenverarbeitungskennntnissen (DV):
- Anfänger
 - gelegentlicher Nutzer
 - Experte
9. Welche Benutzerklassen sollten unterschieden werden?
- Kriterien zur Unterscheidung von Benutzern
- **Vorwissen/Fachliche Erfahrung:** zum Anwendungsgebiet -> mit Erfahrung geht Anwender mit Erwartungen an neues Tool
 - **Vertrautheit:** mit Begriffen und Konzepten der DV prägen das Verhalten beim Dialog
 - **Kognitive Fähigkeiten:** Stil der Problemlösung, Experimentierfreudigkeit oder Ängstlichkeit beschleunigen oder verzögern das Lernverhalten
 - **Einstellungen:** ggüber Dialogsystemen können Effektivität beeinflussen -> grundsätzliche Haltung ggüber Computern; persönliche Gründe für das Arbeiten (freiwillig - zwang; privat - beruflich)
 - **Persönliche Ziele:** beim Erlernen der Bedienung des Systems -> vorrangig ist Arbeitsaufgabe, Beherrschen der Systeme nur soweit wie nötig ODER primäres Ziel ist Verstehen und Beherrschen des Systems
10. Was sind Personas?
- beschreiben fiktive Benutzer als Stereotyp und Referenzperson zur Analyse und Definition von Anforderungen an interaktive Systeme -> Stereotyp zwingt zum Nachdenken über konkrete statt allgemeine Nutzer
11. Erläutern sie das Model-View-Controller- Modell!
- entstanden als Experiment aus Seeheim Modell
 - Aufteilung in Klassen - MVC halt..
 - Vorteile: problemlose Integration in OO Entwurf; problemlose Anpassung an unterschiedliche Benutzer; Unterstützung unterschiedlicher Sichten auf Daten

- Probleme: MVC als alleiniges Architekturmodell nicht geeignet; Trennung Ctrl - View oft nicht zweckmäßig

12. Wozu dienen GOMS-Modelle?

- **Goals:** eine Menge von Zielen
- **Operators:** Menge von Operationen
- **Methods:** Menge von Methoden zum Erreichen von Zielen
- **Selection rules:** Menge von Auswahlregeln von Methoden
- geht von geübten Nutzern aus, d.h. jene die das System bereits kennen
- Modell erlaubt Abschätzungen oder Vorhersagen des Lernaufwandes für den Erwerb des prozeduralen Benutzerwissens
- zusätzliche Vorhersagen für Ausführungsaufwand und Performanz
- Kritik: Berücksichtigt nicht Ermüdung des Nutzers, soziales organisatorisches Umfeld, Unberechenbarkeit des Nutzers, Gewohnheiten/Vorlieben des Nutzers

2.3 Wahrnehmung

1. Welche Prozesse spielen beim statischen und dynamischen Sehen eine Rolle?

- Statisches Sehen: Hell-Dunkel, Farbe, Scharf Sehen; nicht sehen
- dynamisches Sehen: Bewegungen; Peripheres sehen, Augenbewegungen
- Informationswahrnehmung
- Hell-Dunkel: Verantwortlich dafür Stäbchen im Auge -> erweiterbar durch Irisöffnung (schließen bei Helligkeit und vice-versa); Anpassung dauert Sekunden bis Minuten
- nichtSehen: blinder Fleck an Verbindung Sehnerv mit Netzhaut
- Farbe: elektromagnetische Wellen wahrnehmen
unterschiedliche Sensibilität für verschiedene Farben
- Peripher Sehen: unscharfes Sehen in der Peripherie der Netzhaut;

2. Wie erfolgt die Wahrnehmung von Objekten beim Menschen?

- Abbildung von 2D Helligkeitsverteilungen auf der Netzhaut in 3D Repräsentation von Objekten
- 80% aller Informationen übers Auge zum Gehirn
- Objekte ohne Mühe wahrnehmen
- Entdecken von Signalen vor einem Rauschhintergrund
- Erkennen Szenen (komplexe Reizkonfiguration)
- Konzentration auf wesentliche Informationen

3. Nennen Sie Faktoren zur Erkennung räumlicher Tiefe!

- monokulare Faktoren: Größendistanz, Überdeckungen, Schattierungen, Helligkeit und Kontrast
- Bewegungsfaktoren: Bewegungsparallaxe (=scheinbare Bewegung eines Objektes, wenn man seine eigene Position verändert); kinematische Tiefeneindrücke
- binokulare Faktoren: Binokuläre Disparität -> sehen Bild mit jedem Auge, aber durch Entfernung leicht unterschiedlich -> Gehirn mischt daraus die Tiefeninformation
- okulomotorische Faktoren: Konvergenz (Drehung der Augen) -> Bsp: einwärts Drehen um dichte Objekte scharf zu stellen; Akkomodation (=Formveränderung der Linse)

4. Wie erkennen wir Text?

- Verarbeitung von 4 Zeichen links und 6 Zeichen rechts vom Fixationspunkt (europ. Raum)
- Außerhalb dieses Bereiches nur Wortmerkmale (Form, Trennzeichen, Umhüllung) zur Bestimmung des Fixationspunkts
- zeitliche Auflösung beim Lesen: 50 - 200ms
- nach Studie: wichtig sind nur erster und letzter Buchstabe; Reihenfolge der inneren egal -> Gehirn sortiert diese richtig

5. Erläutern Sie die Gestaltgesetze von Wertheimer!

- "Das Ganze ist mehr als die Summe seiner Teile!"
- Prinzipien der Gestaltbildung: Ausnutzung strukturell informationstragender Teile (Wende-, Knick-, Krümmungspunkte), weglassen redundanter Teile
- Gesetze:
 - Figur und Grund: (-Trennung) helle, symmetrische, konvexe oder kleine Flächen zur Figur; dunkle, asymmetrische, konkave oder größere Flächen werden zum Hintergrund (oftmals)
 - Gesetz der guten Form (Gestalt): Betrachter bildet Gruppen von Darstellungselementen aufgrund der Neigung zur Einfachheit, Regelmäßigkeit, innerem Gleichgewicht, Symmetrie und Geschlossenheit von Formen
 - Gesetz der Ähnlichkeit: Tendenz zur Gruppierung von Darstellungselementen gleicher Art (Größe, Form, Farbe)
 - Gesetz der Geschlossenheit: Objekte werden so wahrgenommen, dass sich eine einfache Interpretation ergibt
 - Gesetz der Erfahrung/Erwartung: Bedeutungsinhalt hilft Figuren vom Hintergrund zu trennen

- Gesetz der Kontinuität
6. Was beinhalten die 3 Farbtheorie und die Gegenfarbtheorie?
 - 3Farbtheorie: es existieren 3 Rezeptorsysteme für unterschiedliche spektrale Empfindlichkeit
 - Gegenfarbtheorie: Gegensatzpaare: Rot-Grün; Blau-Gelb
 7. Wie lässt sich Farbe beim Entwurf von Benutzungsschnittstellen einsetzen?
 - Farben harmonisieren oder erzeugen Disharmonie (erzeugt Ablehnung)
 - Unterstützung der visuellen Informationsverarbeitung
 - kann vom Nutzer als ästhetisch ansprechender, hilfreicher empfunden werden
 - können subjektive Sicherheit erhöhen
 - Farbe unterstützt Kommunikation mit Empfänger, wenn Farbbregeln des Anwendungsgebietes eingehalten werden
 - bei gelungener Farbgestaltung: bessere Informationsverarbeitung
 - Visualisieren von Zuständen/Übergängen/Unterschieden
 - Lenken der Aufmerksamkeit auf wichtige Inhalte
 - Markieren selektierter Elemente
 - trennen von Informationskategorien
 8. Was ist bei der Auswahl von Farben zu beachten?
 - Farbkontrast: Bsp: gleiche Farbe wird vor dunklem Hintergrund heller als vor hellem Hintergrund; Warm-Kalt-Kontrast
 - Farbharmonie
 - Farbklänge: Kombination aus mehreren Farben zur Darstellung unterschiedlicher Sachverhalte
 - Farbassoziationen: Bsp - Rot = Blut, Feuer, Gefahr
 - Farbwirkungen unterschiedlich in verschiedenen Kulturkreisen (Bsp: Rot -> Frankreich=Adel; USA=Gefahr) und unterschiedlich in Berufen (Rot - Ingenieure=Gefahr; Finanzmanager=unprofitabel)
 - Farbige Darstellungen haben eine zusätzliche Dimension (ggüber monochromatischen)
 - weniger ist mehr -> besser wenige, aufeinander abgestimmte Farben nehmen
 - einheitliche Farbgestaltung innerhalb eines Kontextes
 - wichtiges durch Farbkontrast hervorheben
 9. Wozu sollten Farben verwendet werden?

- Eine Komposition aus Farben, die miteinander harmonisieren, führt zu einem positiven Gesamtbild.
 - Unterscheidung von Elementen durch reine, gesättigte Farbtöne
 - klare, gesättigte Farben eher für kleine Flächen
 - helle oder entsättigte Farben für große Flächen
 - kräftige bis dunkle Farben eignen sich besonders gut für Schrift, Linien und Strichzeichnungen
10. Welche Sinneseindrücke lassen sich neben der visuellen Wahrnehmung in interaktiven Systemen einsetzen?
- Einsatz von Sprache bzw. Audiosignalen:
 - kleine Ausgabefläche -> wenig Platz für Interaktionselemente (Bsp: Siri)
 - Sprachausgabe technisch relativ leicht zu realisieren
 - Aufmerksamkeit erregen bei Überwachungssysteme (erzeugt mehr Aufmerksamkeit als Visuell)

2.4 Gedächtnis und Handlungsprozesse

1. Was besagt das MultispeichermodeLL für das Gedächtnis?
 - 3 grundlegende, unabhängige Speicherarten:
 - Sensorischer Speicher: sehr kurze Lebensdauer (ms bis s), sehr hohe Kapazität, Auffrischen nur durch Wiederholung mgl
 - Kurzzeitspeicher: bewusste Kontrollverarbeitung; begrenzte Kapazität; Komplexität von Kodierung abhängig (IP vs Domain Name); speichert vor allem symbolische Daten; Lebensdauer wenige Sekunden bis einige Minuten
 - Langzeitgedächtnis: unbegrenzter Speicher; langsame Zugriffszeit (min 0.1s); Speicherdauer Minuten bis Jahrzehnte - abhängig von Qualität und Intensität des Einprägens: Erinnern führt zur Aktivierung von anderen Einheiten
Inhalt: Fakten, Konzepte, Bilder, Vergleiche, Wissen, Fähigkeiten und Abläufe
 - Konsequenzen bei Gestaltung der UI: Kurzzeitgedächtnis -> Informationen gruppieren, Metaphern anwenden, möglichst wenig Ablenkungen
Langzeit: Konsistenz von Kommandosprachen und Menühierarchien durch Strukturierung
2. Welche Schlussfolgerungen ergeben sich aus den Experimenten von Santa?
 - Ergebnisse:
 - Geometrische Formen im visuellen Kurzzeitgedächtnis in räumlicher Anordnung; Erfordern Recodierung, wenn räumlicher Prüfreiz abweicht
 - Wörter im Kurzzeitgedächtnis unter Verlust der Ortsinformation als Kette gespeichert

- Schlussfolgerungen:
- Objekt mit mehreren verbalen oder verbal codierten Elementen in Reihen (Zeilen, Spalten) anordnen
- Objekt aus grafischen Elementen -> räumliche Anordnung bei jedem Vorkommen sollte gleich sein
-

3. Was besagen die Gesetze von Hick und Fitt?

- Fitt: Zeit zur Bewegung auf das Ziel abhängig von Entfernung und Größe des Ziels (+Konstanten Skalierung, Suchzeit)
- Hick: Zeit zur Auswahl eines Ziels abhängig von Anzahl der Alternativen +(konstanten wie bei Fitt)
- Schlussfolgerungen:
- Fitt: Ziele nicht zu klein darstellen, Entfernung bei nachfolgenden Schritten minimieren
- aus komplexen Alternativen auswählen kostet mehr Zeit
- aus mehr Alternativen gleichzeitig auswählen geht schneller als aus einer verschachtelten Auswahl von wenigen Alternativen

4. Nennen sie Phasen von Bedienhandlungen!

- Entscheiden, was zu tun
- Formulieren einer Absicht
- Spezifikation einer (Bedien-)Handlung
- Ausführen einer Bedienhandlung
- Wahrnehmung der Reaktion des Systems
- Interpretation des Systemzustandes
- Vergleich zwischen interpretieren Systemzustand und ursprünglichem Ziel

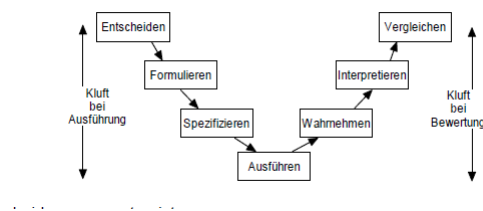


Abbildung 6: Phasen bei Bedienhandlungen

Kluft der Ausführung = Gulf of Execution; anderes = Gulf of Evaluation (siehe IE)

5. Welche Arten von Handlungsregulation werden unterschieden?
 - Bewusst (interlektuelle Ebene): Nutzung des deklarativen Gedächtnisses, nur eine bewusste interlektuelle Handlung gleichzeitig
 - Routinehandlung (Ebene der flexiblen Handlungsmuster): automatisiert aus Produktionsgedächtnis; wenige parallel mgl; Auslösung willentlich oder situationsbedingt
 - Hochautomatisiert (Sensomotorische Ebene): völlig unbewusst, Regelung aus Sinneswahrnehmung und Motorik
6. Wie erfolgt der Umgang mit Fehlern bei Bedienhandlungen?
 - Fehler = Abweichung vom Handlungsprozess vom beabsichtigtem Verlauf
 - Fehlerentdeckung: Verdacht der Abweichung; Entdeckung der Abweichung; Wahrnehmung der Inkongruenz
 - Fehlerdiagnose: Vergleich mit richtigem Verlauf; Fehlerhypothesen; Interaktionshistorie
 - Fehlerkorrektur: Direkte Korrektur; Kompensation des Fehlers; Stornierung/Undo
 - Fehlervermeidung: Minimierung von Gewohnheits-, Bewegungsfehlern durch Hilfen, Lernprogramme, Konzentration auf die Aufgabe
Abfangen von Fehlern durch syntaktische Prüfungen
Vermeidungsstrategien: zb Einschränkung von Freiheitsgraden; Sicherheitsnachfragen
7. Welche Fehlerebenen werden unterschieden?
 - Interlektuell: bei Zielen: Denkfehler (zb nicht beachten von Nebenwirkungen)
bei Ausführungsüberwachung: Merk und Vergessensfehler
bei Rückmeldung: Urteilsfehler
 - Ebene der Handlungsmuster: bei Zielen: Gewohnheitsfehler
bei Ausführungsüberwachung: Unterlassensfehler (zb Überspringen)
bei Rückmeldung: Erkennensfehler (Übersehen, Überhören)
 - Sensomotorisch: in allen Handlungsprozessen - Bewegungsfehler (Vertippen, Verklicken,...)
8. Welche Codierungsformen können für die Darstellung von Informationen eingesetzt werden?
 - sonstiges: (nicht zwingend hierfür relevant, glaub ich)
 - Hervorheben: Betonung des Rands; Variation der Größe, Isolation,..
 - besondere Farbe oder Intensität
 - besonderes Texturmerkmal
 - Blinken oder Bewegung

Codierungsform	Stufenanzahl	Unterscheidbarkeit
Symbol	beliebig	ausgezeichnet
bildliche Formen	10	gut
Position	9	gut
Winkel	8	gut
Farbton	6	gut
Länge	6	gut
geometrische Form	5	gut
Fläche	3	gering
Schriftgröße	3	gering
Linienart	3	gering
Schraffur	3	gering
Schriftformen	3	gering
Blinkfrequenz	2	gering
Helligkeit	2	gering

Abbildung 7: Codierungsformen zur Informationsdarstellung

9. Welche charakteristischen Eigenschaften sollten zur Darstellung von Informationen nach ISO 9241 (Teil 12) beachtet werden?

- Klarheit (schnelle und genaue Vermittlungen des Informationsgehalts)
- Unterscheidbarkeit (dargestellte Information kann genau unterschieden werden)
- Kompaktheit (nur Informationen die notwendig sind)
- Konsistenz (gleiche Information wird stets gleich dargestellt)
- Erkennbarkeit (Aufmerksamkeit des Nutzers auf Information lenken)
- Lesbarkeit (Information ist gleich zu lesen)
- Verständlichkeit (Bedeutung ist leicht verständlich, eindeutig, interpretierbar und erkennbar)

10. Was ist beim Einsatz von Textfonts zu beachten?

- ausreichend groß und gut lesbar
- keine Kursivschreibung (Probleme bei schrägen Linien bei niedriger Auflösung) und keine Großschreibung(d.h. ALLE BUCHSTABEN GROSS :-))
- optimal: Serifenschrift
- Zeilenlänge = 20fache der Schriftgröße
- sinnvoller Zeilenabstand
- nach ISO 9241 (Auszug)
- Zeichenbreite abhängig von Zeichenhöhe
- Wortabstand mindestens ein Zeichen groß
- max Anzahl Zeichen pro Zeile = 60

11. Wozu lässt sich Gruppierung nutzen und wie kann sie umgesetzt werden?

- Elemente mit Sinnzusammenhang in Gruppen zusammenfassen

- Elemente mit ähnlichem Aussehen
- Elemente in Gruppe entsprechend des Arbeitsablaufes darstellen
- für Suchen und Vergleichen: Elemente der Gruppe in Spalte anordnen
- Überschriften und Rahmen erhöhen Übersicht
- Umsetzung: Gruppierung nach fachlichen Kriterien
- zusammengehörige Informationen auf Maske sollten deutlich, auch räumlich, zusammen angeordnet sein
- Fluchtlinien minimieren -> Ausrichtung aller Elemente an wenigen Fluchtlinien

2.5 Normen und Bildschirmarbeit

1. Was beinhaltet ISO 9241-Teil 110 Grundsätze ergonomischer Dialoggestaltung?
 - Dialogrichtlinien
 - unabhängig von bestimmten Dialogtechniken
 - liefert ergonomische Gestaltung von interaktiven Systemen und
 - *Grundsätze der Dialoggestaltung:*
 - Aufgabenangemessenheit: System unterstützt User beim Erledigen seiner Aufgabe, d.h. Funktionalität ist an Aufgabenbereich angepasst
 - Selbstbeschreibungsfähigkeit: muss für Nutzer offensichtlich sein zu jeder Zeit zu erkennen wo er sich im Dialog befindet, welche Handlungen unternommen werden können
 - Steuerbarkeit: Dialogablauf: starten, Richtung und Geschwindigkeit beeinflussbar durch User
 - Erwartungskonformität: entspricht allgemein anerkannten Konventionen und aus Nutzerkontext vorhersehbare Benutzerbelange
 - Fehlertoleranz: Arbeitsergebnis kann trotz erkennbar fehlerhafter Eingaben mit minimalen Aufwand seitens des Benutzers erreicht werden
 - Individualisierbarkeit: HCI und Darstellung von Informationen kann durch Nutzer angepasst werden
 - Lernförderlichkeit: unterstützt Nutzer beim Erlernen des Systems
- Anwendung der Grundsätze unter Berücksichtigung der Benutzermerkmale: Aufmerksamkeitsspanne; Grenzen des Kurzzeitgedächtnisses
Lerngewohnheiten, Grad an Erfahrung bzgl Arbeit und Umgang mit Dialogsystems
mentales Modell des Nutzers bzgl des Dialogsystems

- Grundsätze sind nicht unabhängig voneinander -> Priorisieren der Grundsätze nach Zielen der Organisation; Benutzerbelange; zu unterstützende Aufgaben
2. Wie erreiche ich ...
- Selbsterklärungsfähigkeit? angezeigte Informationen sollten handlungsbegleitend sein; minimierte Notwendigkeit von Handbüchern (oder anderen externen Hilfen); Informierung über Änderung des Zustands; Überblick über die nächsten Dialogschritte;
 - Aufgabenangemessenheit? zeige nur Informationen an, die im Zusammenhang mit der Aufgabe stehen (Bsp: Einhaltung zeitkritischer Bearbeitung -> einzuhaltende Termine zuerst anzeigen); Form der Ein/Ausgabe sollte zu Aufgabe passen; Dialogschritte sollten zum Arbeitsablauf passen
 - Steuerbarkeit? Geschwindigkeit durch Nutzer und nicht durchs System vorgeben; Steuerung der anzuzeigenden Datenmenge; Unterbrechen und Wiederaufnehmen des Dialogs ermöglichen; Elemente zur Navigation durch den Dialog bereitstellen
 - Erwartungskonformität? Vokabular der Aufgabe verwenden; Verhalten und Darstellung innerhalb einer Aufgabe/ähnlichen Aufgaben konsistent; Formate an kulturelle und sprachliche Konventionen anpassen -> Bsp: Symbole für Öffnen, Neu, Speichern -> gleiches Verhalten wie in bekannten Tools
 - Fehlertoleranz? Fehler dem Nutzer anzeigen; Verbesserungsvorschläge anbieten; über automatische Korrektur von Fehler informieren; Überprüfung der Eingabe von Fehlern VOR Verarbeitung; Verhinderung von Systemabstürzen
 - Individualisierbarkeit? Techniken zur Anpassung an charakteristische Eigenschaften des Nutzers; zb Farbschema auswählen; Umfang von Erläuterungen (zb Hilfeinformationen) an Nutzer bzw dessen Wissen anpassbar; Format und Typ der Ein/Ausgabe änderbar; Reset auf Werkseinstellungen mgl;
 - Lernförderlichkeit? Rückmeldungen und Erläuterungen nutzen; minimale Eingabe von Informationen, aber zusätzliche Informationen auf Anforderung liefern; Regeln und zugrundeliegende Konzepte dem Nutzer zugänglich machen
3. Wie lässt sich Gebrauchstauglichkeit messen?
in 4 Aspekten
- **Effektivität:** Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen
 - **Effizienz:** Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand zum Erreichen des Ziels
 - **Zufriedenstellung:** Freiheit von Beeinträchtigungen und positive Einstellungen ggüber der Nutzung des Produkts
 - **Nutzungskontext:** Benutzer, Arbeitsaufgaben, Arbeitsmittel, physische und soziale Umgebung

- 1-3 werden im Rahmen des Nutzungskontextes betrachtet
4. Nennen sie Kriterien zur Messung der Gebrauchstauglichkeit!
siehe eine Frage vorher(?)

5. Was lässt sich messen bei der Gebrauchstauglichkeit? (allgemein)

Maß der Effektivität	Maß der Effizienz	Maß der Zufriedenstellung
Grad der Zielerreichung in %	Zeit für Erledigung einer Aufgabe	Einstufungsskala der Zufriedenheit
%-Satz der Nutzer die Aufgabe erfolgreich abschließen	Abgeschlossene Aufgaben je Zeiteinheit	Häufigkeit freiwilliger Nutzung
Durchs. Genauigkeit der abgeschlossenen Aufgaben	Monetäre Kosten der Aufgabenerledigung	Häufigkeit von Beschwerden

6. Ein Dialogsystem soll angemessen für geübte Benutzer sein! Welche Usability- Messungen sind sinnvoll?

Maß der Effektivität	Maß der Effizienz	Maß der Zufriedenstellung
Anzahl ausgeführter Aufgaben hoher Schwierigkeit	Effizienz im Verhältnis zu einem Experten-Benutzer	Einstufungsskala für Zufriedenstellung mit Produktmerkmalen für hohe Ansprüche
Prozentsatz relevanter Funktionen, die genutzt werden		

7. Wie kann der Nachweis der Gebrauchstauglichkeit für Software geführt werden?

- mit dem ErgoNorm-Prüfverfahren; Verifikation sehr teuer, daher Falsifizierungsansatz (Software ist richtig bis Gegenbeweis)
- zweiteiliges Prüfverfahren:
 - a) Benutzerfragebogen -> Untersuchung von Nutzerproblemen
 - b) Prüfverfahren: Anforderungsentwicklung und Normkonformitätstest sowie Erhärtungsprüfung für vermutete Normabweichungen -> Methoden: Aufgabenanalyse mittels Szenarien; Inspektion; Dokumentanalyse,...
- Definition von Prüfkriterien: Nutzungskontext —erkennen der—> Erfordernisse —ableiten der—> Anforderungen —transformieren in—> Prüfkriterien

- **Erhärungsprüfung:**

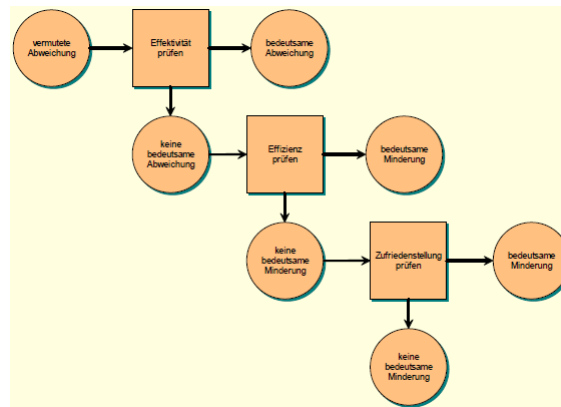


Abbildung 8: Entscheidungsregeln bei Erhärungsprüfung

- **Nonkonformitätsprüfung:**

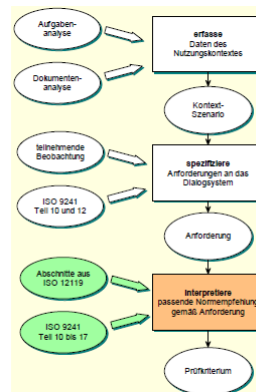


Abbildung 9: Nonkonformitätsprüfung

– ISO Teil 10 Dialogprinzipien; Teil 12 Informationsdarstellung

8. Was regelt die Bildschirmarbeitsverordnung?

- Vorschriften zur Vorsorge, zum Arbeitsablauf und zu Pausen für alle Beschäftigten
- gilt nicht für öffentliche EDV Geräte; Rechenmaschinen, Kassen mit kleiner Datenanzeige, Bildschirme für Videoüberwachung,....
- Mindestanforderungen für Arbeitsplatzgestaltung:
Arbeitsmittel: Bildschirm, Tastatur, Tisch und Stuhl
Arbeitsumgebung: Bewegungsraum, Licht, Lärm, Klima, Strahlung
Zusammenwirken Mensch-Arbeitsmittel: Software und Arbeitsaufgaben

- Anforderungen an Körperhaltung:
optimaler Abstand zum Gerät: 90cm; optimale Blickrichtung
großer Greifraum
 - Arbeitstätigkeit am Bildschirm regelmäßig durch andere Tätigkeiten oder Pausen unterbrechen
 - Prüfsiegel für Geräte
 - Anforderungen an Bildschirme:
Ergonomie: gleichmäßige Beleuchtung; Kontrast; Flimmerfrei,..
Emissionen: magnetische Störungen dürfen kein Flackern verursachen
Ökologie: geringer Energieverbrauch; Recyclebar,...
9. Wie kann die Einhaltung von Ergonomie-Normen geprüft werden?
müsste zwei Fragen vorher das ErgoNorm Verfahren sein

2.6 Fenstersysteme

1. Welche Begriffe/Konzepte werden neben Fenstern bei modernen Fenstersystemen noch benötigt?
 - Window: Ausgabefläche, die zu platzieren ist
 - Desktop/Workspace: Fläche auf dem Fenster platziert werden
 - Display: Verwaltung der Ausgabefläche und Eingabegeräte
 - Screen: physisch vorhandener Bildschirm
 - Virtual Screen/Workspace/Desktop: Durch Speicherabbildung vorhandener Platz aus dem aktuelle Ausschnitte eingeblendet werden können
2. Fenstersysteme – Strategien zur Platzierung von Fenstern
 - **Compositing:** Fenster werden erst erstellt, separat gemalt und dann zusammengepackt um sie in verschiedenen 2D oder 3D Umgebungen darzustellen
 - **Stacking:** Fenster können überlappend dargestellt werden indem Hintergrundfenster zuerst gezeichnet werden; darf nicht als Composition Manager deklariert sein
 - **Tilling:** nicht überlappend; Fenster werden neben- oder unter/übereinander dargestellt
 - **Dynamic:** switchen dynamisch zw. stacking und tilling

3. Erläutern Sie das Schichtenmodell eines Fenstersystems!

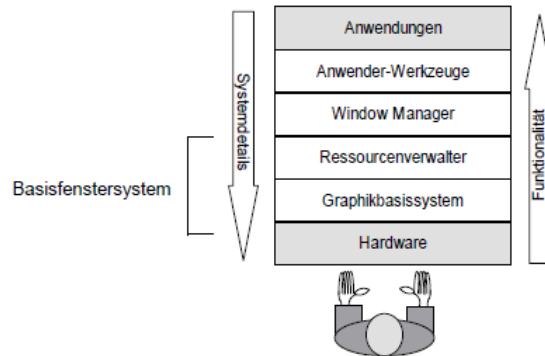


Abbildung 10: Schichtenmodell eines Fenstersystems

- **Anwenderwerkzeuge**

- Typische Dialogbausteine müssen nicht für jede Applikation neu implementiert werden (Toolkits)
- Schnittstelle zw Anwendung und Fenstersystemfunktionalität
- typischerweise sind Bibliotheken die Anwenderwerkzeuge

- **Window Management**

- Anzeige verschiedener Applikationen auf dem Bildschirm; Dialogmöglichkeiten auf Sitzungsebene
- zuständig für Aussehen und Verhalten
- sorgt für Positionierung und die Ausstattung der Fenster
- liefert Standardtechniken, mit denen der Benutzer interagieren kann (zb Close Button)
- Bildschirmaufteilung: Wo ist noch Platz? Welche Anwendung ist ikonifiziert?
- Sitzungsverwaltung: Navigation zw. Applikationen; Menutechnik; Fensterrahmen,...
- verschiedene Window Manager: siehe Frage vorher
-

- **Ressourcen Verwaltung**

- Konkurrierender Zugriff auf Ressourcen; Synchronisation zw Applikationen
- für k Terminals (Monitor, Eingabegeräte) gibt es genau eine Instanz eines Fenstersystems; Dazu n Anwendungen mit je m Fenstern

- Elementare Konzepte:
- Objekt "Fenster": Zustand (Sichtbarkeit, (in)aktiv); Größe; Fußpunkt (Darstellung auf Screen)
- Objekt "Ereignis": Eventtyp; Timestamp; Ort; zugehöriges Fenster
- Objekt "Font"
- Objekt "Graphikkontext"
- Objekt "Farbtabelle"

- **Graphikbasissystem**

- Realisiert Ausgabefunktionalität für Fenstersysteme (2D-Rastergrafik); hardwarenah angelegt für hohe Ausgabegeschwindigkeit
- Elementare Konzepte - 2D:
- Zeichenfläche - Fenster (Drawable): Ebenenformat - XY Koordinaten; Tiefe: Z-Format
- Ausgabe-Objekte und grafische Kontexte: Linien- oder Flächenprimitive, Textausgabe
- Gedächtnis der Ausgabe: Daten-gepufferte Ausgabe (Ausgabe in Hintergrundspeicher, anschließend Teile auf Bildschirm kopieren)
Bearbeiten und Vergessen: Rasterkonvertierung und dann Ausgabe
- Events und Warteschlange: Umwandlung gerätespez. Events und Speicherung in Queue (log. Bildkoordinaten, Timestamp, Eventcode)

- **Hardware**

- Rasterbildschirm mit Grafikprozessor, Maus, Tastatur, Multimediakomponenten

4. Was sind Widgets und nennen sie typische Elemente in Widgetbibliotheken?

- Widget = Dialogbaustein (heute auch kleine Anwendungen bei Android/Win 7)
- in Motif:
- Primitive Widgets: Dialogbausteine mit Darstellung und Interaktion (Buttons,...)
- Composite Widgets: Dialogbausteine zur Anordnung von Elementen (Layout-Manager)
- Shell Widgets: Dialogbausteine zum Behandeln neuer Fenster (Fenster, Dialogboxen)

5. Welche Probleme treten bei der Anbindung von Funktionalität in objektorientierten Sprachen auf?

- Implementierung oft in C (da OS-nah)
 - Fenstersystem vom Prinzip her aber OO
 - Probleme:
 - Widgets: Widget in C++ Wrapper einschließen
Problem: C++Instanz ex. bevor Widget dargestellt wird; Unklar wann Ressourcen in Unterklassen das Widget beeinflussen; Eventanbindung
 - Eventanbindung: Events durch Callbacks realisiert, CB benötigt Referenz auf Widget (siehe oben Wrapper vor Widget initialisiert)
Lösung: 1. CB als globale Funktion;
2. statische Memberfunktionen
6. Welche Vorteil bietet "Markup and Code-Behind" gegenüber herkömmlicher Benutzungsoberflächenprogrammierung?
- Markup+Code-Beding: Trennung der Oberfläche und Anwendungslogik
 - dadurch können Designer und Entwickler parallel arbeiten
 - einfacher Austausch der Komponenten
7. Nennen Sie Bibliotheken zur Gestaltung graphischer Benutzungsoberflächen und vergleichen sie diese! weitere: Java -> AWT; Swing;SWT

GTK+	QT	WPF
Signals and Slots als Event-handling	Signals and Slots als Event-handling	
in C mit C++Wrapper für OO; vorrangig für X-Window	plattformübergreifend; C++	für Windows ab Vista; setzt auf DirectX
besteht aus Teilbibliotheken		Oberfläche 2D, Fenster Ikonen in 3D
		Markup and Code Behind

2.7 Dialogtechniken

- Was sind Interaktions- und Dialogtechniken?
 - Interaktionstechnik = Realisierung einer Interaktionsaufgabe
Charakterisierung durch die zu lösende Aufgabe und zu welchem Interaktionsstil sie angehören
Interaktionsaufgabe = bestimmte Aktion, die wiederholt oder in ähnlicher Form auftreten
 - wiederkehrende Dialogmuster durch sog. Interaktionsformen beschreiben und realisieren
Interaktionsform = festgelegt Auswahl von Ein/Ausgabevorgängen zsm mit ihren Abfolgebedingungen, die zu stereotypischen Kommunikationssequenzen führen

- *typische Techniken:*
 - natürliche Sprache: Anwendung in KI, Syntaktische und Semantische Analyse; Sprachein/ausgabe
 - Frage-Antwort-Dialoge: System fragt; Nutzer antwortet
 - Kommandodialog: veränderter Frage-Antwort; Nutzer stellt System jetzt Kommandos (klassisch bash ;-))
 - Menügesteuerter Dialog:
 - Formulardialog: quasi nachgebildetes Papierformular
 - Direkte Manipulation
 - Graphischer Dialog
 - Fenstertechnik - WYSIWYG
 - heute durch Fenstersysteme: Kopplung unterschiedlicher Stile
- Vergleichen Sie PullDown mit Popup-Menüs! Standardelemente des Fenstersystems

	Pulldown	Popup
Bsp	klappt unter Titel auf nachdem Titel angeklickt wurde; verschwindet nach Auswahl oder bewegen des Cursors außerhalb der Menufläche klassische Menuleiste	unsichtbares Menu; erscheint an Cursorposition Contextmenu
Vorteile	Darstellung: ständiger Hinweis auf Steuerungsmglkeiten durch Menuleiste Wirkungsbreite: Optionen können über mehrere Bildschirmbereiche wirken Raumbedarf: relativ gering aufm Bildschirm	Mauseinsatz: Minimierung der erforderlichen Mausebewegung Raumbedarf: Nutzfläche bleibt vollständig erhalten
Nachteile	Optionenmenge: nur beschränkte Auswahl von Optionen mgl Mauseinsatz: längere Mausebewegungen erforderlich	Optionenmenge: nur geringe Anzahl von Optionen mgl Wirkungsbreite: Aktionen sind objektbezogen Darstellung: Steuerungsmglkeiten erst nach Öffnen sichtbar

- Beschreiben Sie Varianten von Menüs!
 - Binäres Menü: genau zwei auswählbare Elemente
 - Erweitertes Menü: Menu in Seiten untergliedert
 - Einfach/Mehrfac-Menu: Nutzer kann genau/mehr als eine Wahl treffen

- Permanentes Menu: ständig auf Bildschirm sichtbar
- Pulldown/Popup: siehe oben
- Drop-Down: Menu klappt unter Titel auf nachdem der Titel berührt wurde; verschwindet auch wieder
- Tear-Off:

- Welche Vor- und Nachteile haben spezielle Dialogtechniken (Sprache, Menü, Formular)?

	Natürliche Sprache	Menü	Formular
Vorteile	einfach anwendbar mächtig genug, um jede Anfrage zu übermitteln keine Formale Sprache erlernen	vollständige Führung des Nutzers durchs System einfacher Auswahlmechanismus schnell bei kleinen Wertemengen	explizit Werte in Maske eingetragen Navigation zw Eingabefeldern Vertrautheit bei Formularen, die Papier entsprechen
Nachteile	derzeit nur Verarbeitung von Teilmenge der Sprache Benutzer überschätzen Fähigkeiten des Systems Komplexe Sachverhalte oft nur umständlich zu formulieren oft Mehrdeutigkeiten	langatmiger Dialog bei großer Auswahlmöglichkeit Mehrdeutigkeit der Zuordnung (wo finde ich was?) Wie lange Menu sichtbar?	Masken oft mit Informationen überladen Probleme, wenn alle Felder ausgefüllt werden müssen Layoutgestaltung (Prinzipien der Wahrnehmungslehre)

- Welche Aufgabe hat ein Layoutmanager!
 - Anordnung der Kindelemente nach bestimmten Kriterien
- Nennen sie typische Beispiele für Layoutmanager!
 - Lineare: Kinder nacheinander in Reihenfolge der Richtung (vertical/horizontal) eingeordnet
 - tabellarisch:
 - Anordnung in Rubriken (zb BorderLayout Java)
 - Anordnung mit Hilfe von Constraints
 - Baum- oder Graphenalgorithmen
- Nennen sie 3D Interaktionstechniken?
 - Trackball (= 3D Hülle um Objekt)
 - Semitransparente Volumina als 3D Cursor
 - Anfassbare physische Objekte (zb Linse auf Tabletop)
- Welche Möglichkeiten hat man zur „Erweiterung“ der Zeichenfläche?
 - Nutzen mehrerer, alternativer Zeichenflächen (Card-Layout, Tab-Layout)
 - Scrollbars (Verschieben des darstellbaren Bereichs)
 - ZUIs
- Was ist direkte Manipulation?

- Prinzipien:
 - permanente Sichtbarkeit der jeweils interessierenden Objekte
 - schnelle, umkehrbare Operationen, deren Wirkung unmittelbar sichtbar werden
 - physische Aktionen, wie Bewegen eines Zeigegerätes, Selektionsmechanismen
 - generische Kommandos und Funktionsobjekte (Mülleimer, Printer)
 - OO Interaktionssyntax: erst Auswahl Objekt, dann Methoden/Kommandoauswahl
- Vorteile:
 - schnell erlernbar (für Anfänger); effizient (für Experten)
 - kaum Fehlermeldungen notwendig
- Probleme:
 - leistungsfähige Zeigeeinstrumente notwendig
 - hoher Programmieraufwand
- Wie würden sie Menüs an einer Powerwall nutzen? Ziele der Interaktion?
PopupMenu mit am sinnvollsten; DropDown mit ShortCuts auch sinnvoll mgl
- Welche Gesten/Interaktionsgeräte sind sinnvoll einsetzbar an Powerwall?
Eyetracker; Tracker für Bewegung des Körpers (Bsp Wii)
- Welche Gesten halten sie für sinnvoll zum Ersatz von „arm“ und „activate“ auf einen Touchscreen?

2.8 Metaphern

- Was ist eine Metapher ?
 - sprachliches Bild
 - Dadurch wird auf die Ähnlichkeit der Begriffe in einem speziellen Kontext hingewiesen
 - Metaphern nutzen die Vertrautheit eines Begriffes, um ein Konzept anschaulich zu machen – positive Analogien.
 - Metaphern können Ordnung und Struktur in Schnittstellen bringen bei Orientierung an Nutzer vertraute Konzepte
Bsp: Desktop-Metapher; Buch-Metapher; Reise-Metapher („erst Aufgabe erledigen bevor ich weiter kann“); Raum-Metapher (zb in Informationssystemen)

- Wie lassen sich Metaphern bei der Arbeit mit Dialogsystemen sinnvoll nutzen?
siehe vorher Punkt 3
- Ein Entwurfsmuster beschreibt ein bestimmtes, in einem bestimmten Kontext immer wiederkehrendes Entwurfsproblem sowie ein generisches Schema zur Lösung dieses Problems.
- Wozu dient das Entwurfsmuster Fassade?
 - einheitliches Interface für Menge von Klassen in einem Subsystem
 - ermöglicht dadurch leichtere Nachnutzung des Subsystems
 - Anwendbarkeit: einfache Schnittstelle zu komplexen Subsystemen (zb WidgetBibliothek)
 - Entkopplung des Systems und Aufteilung in Schichten
- Wozu dient das Entwurfsmuster Kompositum(Composite)?
 - Zusammenfügen von Objekten in baumartigen Strukturen, um Teile-Ganzes-Hierarchie zu repräsentieren
 - Teilnehmer: Client - manipuliert Objekte durch Schnittstelle von Komponente
Blatt: repräsentiert Kindobjekt, definiert Verhalten für primitive Objekte
Komponenten: deklariert Schnittstelle für Objekte in Struktur und Zugriff auf Kindobjekte
 - Kompositum: definiert Verhalten für Komponenten, die Kindobjekte haben können
- Wozu dient das Entwurfsmuster Dekorierer (auch Wrapper)?
 - Dynamisches Hinzufügen zusätzlicher Verantwortlichkeiten
 - flexible Alternative zur Unterklassenbildung existierender Funktionalität bereit
- Wozu dient das Entwurfsmuster Observer?
 - Definition einer 1:n Abhängigkeit zw Objekten um bei Änderung des einen Objektes alle n abhängigen Objekte zu informieren
- Wozu dient das Entwurfsmuster ...?
Singleton, Abstract Factory; Command; Adapter; Bridge; Proxy
- Vergleichen Sie das Entwurfsmuster Observer mit dem Model-View-Controller-Modell!
 - Observer = Pattern | MVC = Architektur
 - Observer kann im MVC benutzt um effizient die Änderung mitzuteilen