

Zusammenfassung Interaction

Philipp Jäcks

19. Februar 2016

Inhaltsverzeichnis

1	Interaction Engineering - Fragenkatalog	2
1.1	Introduction	2
1.2	Basics of Graphics and Interaction Programming	5

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

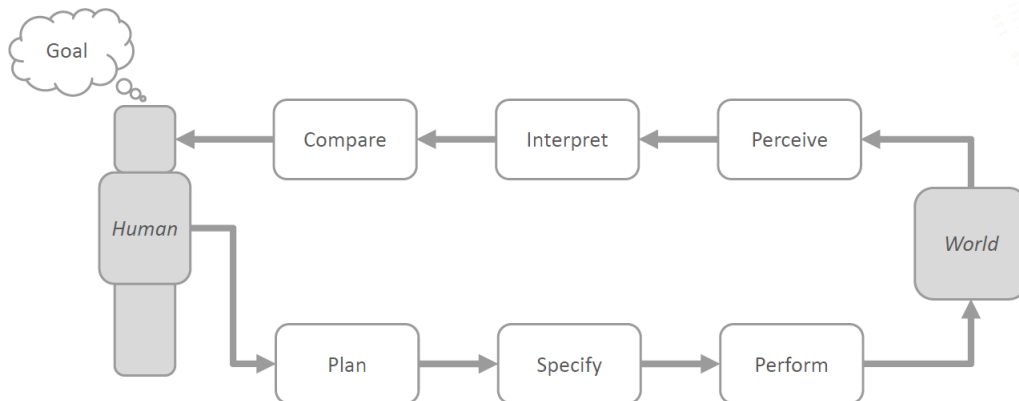


Abbildung 1: Action Cycle nach Norman

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

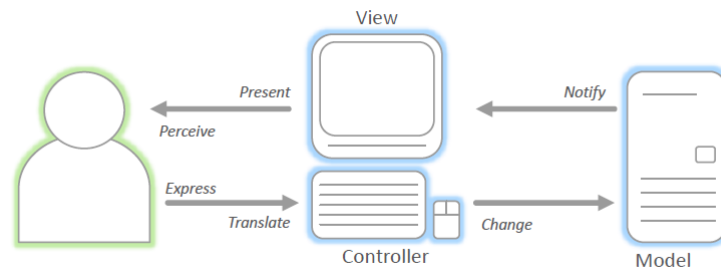


Abbildung 2: Zusammenspiel Action Cycle und MVC

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

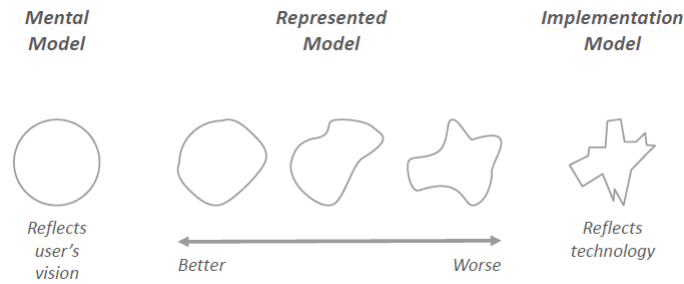


Abbildung 3: Zusammenspiel der 3 Modelle

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!

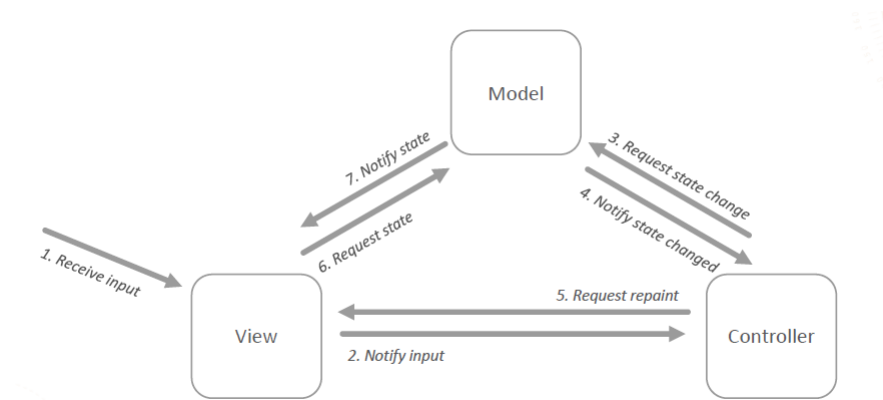


Abbildung 4: Interconnection der drei Bestandteile des MVC Pattern

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