

## Guidelines for Using Multiple Views in Information Visualization

# Richtlinien zur Verwendung mehrerer Ansichten in der Informationvisualisierung

Multiple View Systems:=

Verwenden zwei oder mehr unterschiedliche Ansichten zur Untersuchung einer ‚konzeptiven‘ Einheit. Sie sind weit verbreitet und nützlich, da sie die Möglichkeit bieten Erkennungsprozesse durch gleichzeitiges Betrachten verschiedener Aspekte eines Themas zu beschleunigen.

Multiple View Systems

Bieten eine Vielzahl von Vorteilen: wie verbesserte Benutzer ‚performance‘, das Erkennen unvorhergesehener Zusammenhänge, Vereinheitlichung der Bildschirmoberfläche. Sie stellen aber auch hohe Herausforderungen an den Entwurf: Neben hochentwickelten Koordinationsmechanismen und Darstellungsweisen werden die Entwicklungsentscheidungen durch Interaktionen zwischen den Dimensionen des ‚design space‘ zusätzlich verkompliziert.

Bei der Teilnahme an der MVS-supgroup eines Kongresses ist den Autoren aufgefallen, dass viele unnötige Entwicklungsfehler in MVS gemacht werden. Sie beobachteten Vielschichtigkeiten/Unübersichtlichkeiten und Nicht- Durchgängigkeiten in ihren eigenen Systemen sowie in denen Anderer. Solche Fehler sind besonders schwerwiegend weil sie, aufgrund des komplizierten Aufbaus von MVS, oft schwer zu korrigieren sind.

Die Workshop-Teilnehmer waren der Ansicht, dass der Entwicklungsprozess von Multiple View Systems durch Funktionalitätsüberprüfung nach wissenschaftlichem Ansatz verbessert werden könnte. Da, außer den allgemeinen Entwicklungsrichtlinien für Benutzeroberflächen und den näher verwandten Richtlinien für multimodale Systeme, im Moment wenig spezielle Hilfe verfügbar ist, beschlossen die Autoren ihre gesammelten Erfahrungen als Grundlage einer solchen Richtlinie zu verwenden und zusammenzufassen..

Diese Richtlinie soll der Entwicklung neuer Systeme und der Funktionalitätsüberprüfung /Durchforstung alter Systeme dienen. Teils wurden allgemeine Regeln zu MVS-spezifischen Themen angepaßt, manche Regeln sind einzigartig für MVS. Zur Verdeutlichung werden Beispiele angeführt, wobei aus Platzgründen nicht alle mvs durchgegangen werden.

MVS treten in verschiedenen Variationen auf, von Videospiele über Buchillustrationen bis zu Überwachungskamera-Systemen.

Das vorliegende Paper beschäftigt sich hauptsächlich mit Informationsvisualisierung.

Es gliedert sich in:

- Kosten-Nutzen-Abschätzungen des Entwicklers während der Arbeit an einem MVS
- Definition und Modellierung eines MVS
- Vorstellung von acht Entwicklungs-Richtlinien aufgeteilt in zwei Bereiche:
- Einerseits Richtlinien die dem Entwickler helfen zu entscheiden, ob mehrere Ansichten erstrebenswert sind und welche. Die Einführung von MV ist mit gewissen Kosten verbunden: Komplexität im System verhindern und Wahrnehmungsmehraufwand für den Benutzer.
- Andererseits Richtlinien für die Verwendung von MVS, um den System-Aufwand minimieren.

Diese Richtlinien überschneiden oder widersprechen sich teilweise, in solchen Fällen muß der größere der Nutzen für den Anwender abgeschätzt werden. (große Verantwortung des Entwicklers)

### **Kosten-Nutzen-Betrachtungen**

Die Entscheidung über die Anwendung von MVS für Informationsvisualisierungsprobleme erfordert das Ausbalancieren einer Anzahl von Entwicklungsabwägungen.

Der Nutzen durch die Minimierung des Denkaufwandes für eine einzelne komplexe Ansicht, kann durch den höheren Aufwand für das Springen zwischen Zusammenhängen und durch höhere Systemanforderungen bei mehreren Ansichten herabgesetzt werden.

Auf die Auswirkungen der einzelnen Richtlinien auf Systemanforderungen und kognitiven Aufwand wird explizit eingegangen.

Kognitive Aspekte einer Informationsmanagement-Aufgabe:

- Zeit und Aufwand um das System zu *lernen*
- Die *Last* auf den "Arbeitsspeicher" des Benutzers
- Der für Vergleiche erforderliche Aufwand
- Der für das *Wechseln des Zusammenhanges* erforderliche Aufwand

Durch MVS erzeugter Einfluß auf die Systemanforderungen:

- *Rechnerische Anforderungen* für das Rendern der zusätzlichen Anzeigeelemente
- *Bildschirmplatz* für die zusätzlichen Ansichten

Neben der Abschätzung der Nützlichkeit für den Anwender muß der Designer auch die Ressourcen berücksichtigen, die zur Entwicklung, Einführung und Aufrechterhaltung des Systems erforderlich sind.

### **Modellierung**

3 Dimensionen des MVS:

- *Auswahl* der Ansichten
- *Präsentation* der Ansichten
- *Interaktion* zwischen den Ansichten

Softwareentwickler beginnen notwendigerweise damit, tiefgehendes Verständnis für die Aufgabe des Anwenders zu entwickeln. Danach wird ein System entworfen, das den Nutzer bei der Erfüllung seiner Aufgabe unterstützt. Sind die Nutzer-Bedürfnisse und die System-Architektur festgelegt, kann der Designer das System anhand der Richtlinien entwickeln oder evaluieren. Außerdem hat er eine solide Grundlage für Benutzer-Studien die ihm bei der iterativen Entwicklung helfen.

### **Definition**

Ein Einzelansicht ist eine begriffliche Einheit, aus einem Datensatz samt Festlegung der visuellen Darstellung der Daten. Diese Darstellung kann textlich z.B. als Tabelle oder graphisch z.B. als Balkendiagramm erfolgen.

Unterschiedliche Ansichten ermöglichen dem Nutzer verschiedene Aspekte eines Begriffes zu erfassen, z.B. durch die Präsentation verschiedener Informationen oder durch die Betonung verschiedener Aspekte der gleichen Information. Ein MVS nutzt zwei oder mehrere solcher Ansichten um die Erforschung eines Begriffes zu unterstützen.

Beispiel Power Point: verschiedene Ansichten der Folien um Inhalt bzw. Reihenfolge besser darstellen zu können.

Die Richtlinien sind auf Systeme ausgelegt, bei denen ein Nutzer bestimmte (aufeinander abgestimmte) Ansichten zur Erfüllung einer festgelegten Aufgabe erhält.

Die Ansichtsfenster unterscheiden sich in den dargestellten Daten oder in der Art der Datendarstellung.

- Ein Datensatz kann bereits in einem anderen übergeordneten Datensatz enthalten sein.
- Datensätze können Funktionswerte der Daten eines anderen Datensatzes beinhalten. Restaurantbeispiel - average cost - location
- Datensätze können gänzlich unterschiedliche Informationen beinhalten.

Unabhängig davon ob die Datensätze sich unterscheiden, kann die visuelle Darstellung variieren (Tabellen, Diagramme, etc).

### **Auswahl**

In der ersten Phase des Entwicklungsprozesses erfolgt die Auswahl der Ansichten, die auf koordinierte Weise **der Unterstützung einer bestimmten Aufgabe** dienen sollen.

Wobei zu beachten ist, daß manche Kombinationen bedeutungslos bzw. nicht von Interesse sein können.

### **Präsentation**

- Sequentielle Anzeige
- Simultane Anzeige

Bei gleichzeitiger Anzeige ist die Anordnung auf dem Bildschirm zu bedenken.

### **Interaktion**

Interaktionsmechanismen zwischen den Ansichten.

Jede Einzelansicht hat verschiedene Möglichkeiten (Auswahl, Zoom, Pan,...) die miteinander verbunden sein können, so dass Aktionen in einer Ansicht Auswirkungen auf eine andere Ansicht haben.

- “navigational slaving”: Bewegungen werden automatisch auf die anderen Fenster übertragen
  - “linking” verbindet Daten in verschiedenen Ansichten, speziell : “brushing” dient als Auswahlanzeige durch Hervorheben in den verbundenen Ansichten Restaurantbeispiel - price -location
- slaving und linking werden durch Kopplungsfunktionen erreicht, diese weisen Objekten oder Positionen einer Ansicht, Objekte oder Positionen in einer anderen Ansicht zu.

Sind solche Zuweisungen erfolgt, muß entschieden werden wann sie zum Einsatz kommen. Auch Verbindungen mehrere Ansichten müssen möglich sein.

### Richtlinien

#### **Anwendung von MVS**

Die ersten vier Richtlinien beschäftigen sich mit der Auswahl der Ansichten, sie sollen dem Entwickler bei der Beurteilung helfen, ob MVs für seine Anwendung angebracht ist:

- Vielfalt
- Ergänzung
- Geiz / Sparsamkeit
- Zerlegung

#### **Vielfalt**

*Verwende MV wenn die Daten eine Vielfalt von Eigenschaften, Modellen, Benutzerprofilen, Abstraktionsgraden oder Genres aufweisen.*

Eine einzige Ansicht die vielen Ansprüchen genügen muß, ist oft nur der kleinste gemeinsame Nenner, der für keinen Zweck optimal ist. Dadurch entsteht für der Nutzer großer Aufwand, da er eine große Datenmenge gleichzeitig aufnehmen und verstehen muß, wobei Teile davon für ihn nicht relevant sind.

Das Vorhandensein von Vielfalt ist einer der wichtigsten Gründe ein MVS zu entwickeln.

CAD Beispiel - Schaltkreis, Beispiel Übersicht - Detailansicht

### **Gegenseitige Ergänzung**

*Verwende MV wenn verschiedene Ansichten Zusammenhänge und/oder Ungleichheiten hervorheben.*

Direkter visueller Vergleich ist einfacher als Vergleich aus dem Gedächtnis, durch gleichzeitiges Sehen beeinflussen MVS die Wahrnehmungsfähigkeit und verbessern das Verständnis von Beziehungen zwischen Ansichten. Dadurch helfen MVS komplizierte Zusammenhänge zwischen Datensätzen zu verstehen und sonst verborgene Beziehungen zwischen zwei oder mehreren Ansichten aufzuzeigen.

Beispiel - chemischer Aufbau eines Proteins - Oberfläche, Verbindung ...

### **Zerlegung**

*Teile komplexe Daten in mehrere Ansichten auf um bewältigbare Einheiten zu erhalten und Einsicht in die Interaktionen zwischen den verschiedenen Dimensionen zu gewähren.*

Eine komplexe Einzelansicht kann den Nutzer überfordern. Mehrere Ansichten sind hilfreich indem sie die Datenmenge reduzieren, die gleichzeitig erfasst/verarbeitet werden muß.

Die isolierte Betrachtung von Einzelaspekten eines komplexen Datenobjektes alleine ist schon vorteilhaft, bei verschachtelten Zusammenhängen erleichtert die Zerlegung gemeinsam mit der Regel der Gegenseitigen Ergänzung das Vergleichen.

### **Sparsamkeit**

*Verwende MV so wenig wie möglich.*

Einzelansichten versorgen den Nutzer mit gleichbleibendem Kontext für die Datenanalyse, MV erfordern das Wechseln des Kontextes und führen zu zusätzlicher Kompliziertheit des Systems. Also muß der Entwickler den Aufwand - Lernaufwand Nutzer, rechnerischer und darstellerischer Mehraufwand – durch die vorangegangenen Regeln rechtfertigen (Vielfalt, Ergänzung, Zerlegung)

Mehrere Ansichten fordern erhöhte Aufmerksamkeit vom Nutzer, beanspruchen wertvollen Bildschirmplatz und bedingen außerdem das Lernen mehrerer Gedankengebäude. Deshalb sollten neue Ansichten nur eingefügt werden, wenn berechtigte Gründe dafür bestehen. Falls mehrere Fenster ähnliche Semantik aufweisen, sollte eine Zusammenlegung erwogen werden.

Die Sparsamkeits-Regel sollte auch auf Kopplungen angewendet werden, da diese zur System- Kompliziertheit für Nutzer und Entwickler beitragen. Ein genaues Abwägen des Wertes der Verknüpfung für den Nutzer ist erforderlich.

### **Nutzung von MVS**

Nachdem die Auswahl der anzuzeigenden Ansichten erfolgt ist, muß der Entwickler sich deren Anordnung und ihren Interaktionen widmen.

4 Entwicklungsregeln dienen als Entscheidungshilfe für Entwickler und zur leichteren Auffindung von Problem-Punkten in bestehenden Systemen. für Evaluatoren

- Raum- / Zeit- Ressourcen- Optimierung
- Klarheit (Selbsterklärend)
- Einheitlichkeit
- Aufmerksamkeits-Management

Fehler! **Unbekanntes Schalterargument.**

#### **Raum/Zeit**

*Stelle Gleichgewicht zwischen den räumlichen und zeitlichen Kosten der Präsentation mehrerer Ansichten und den räumlichen und zeitlichen Vorteilen der Benutzung mehrerer Ansichten her.*

Hier bleiben oft die Kosten aus Platzbedarf am Bildschirm und Rechenzeit unberücksichtigt, die aus der gleichzeitigen Darstellung von mehreren Ansichten entstehen, ebenso die Zeit die der Nutzer spart, wenn er Vergleich anstellen will und nebeneinanderstehende Ansichten hat.

Eine der ersten Entscheidungen ist deshalb, die Ansichten

- Räumlich nebeneinander oder
- Zeitlich hintereinander darzustellen.

Selbst wenn die Anwendung dem Nutzer diese Entscheidung freistellt, ist eine gute Voreinstellung wichtig, wobei der Entwickler berücksichtigen sollte, wie viel Platz und Zeit dem Nutzer zur Verfügung steht und wie viel Platz und Zeit jede der infrage kommenden Ansichten beansprucht.

Platzkosten sind relativ einfach zu berechnen, Zeitkosten aufwendiger, versteckte Zeitkosten resultieren aus der Zeit für die geistige Umstellung des Nutzers auf eine andere Ansicht und jene Zeit, die zum berechnen und Rendern einer Ansicht benötigt wird.

Beachte auch Download-Zeiten und den Umstand, daß unterschiedliche Endgeräte unterschiedliche Lösungen erfordern (Bildschirmgröße, Arbeitsspeicher).

### **Selbsterklärend**

*Benutze selbsterklärende Stichworte um Zusammenhänge zwischen Ansichten offensichtlicher zu machen.*

Statisch oder dynamisch, visuell oder auditiv, auffallende Hinweise können bei Vergleichen, den Unterschied zwischen Wahrnehmung und Erkenntnis ausmachen.

Zudem werden Systeme mit selbsterklärenden Stichworten schneller erlernt.

Beziehungen zwischen Ansichten zu erkennen kann eine schwierige Aufgabe sein, die in MVS durch auffallende Hinweise erleichtert wird.

Möglichkeiten hierzu sind:

- Änderungen hervorheben
- Räumliche Anordnung
- Gekoppelte Interaktion

Einschränkungen:

Kopplungen bringen Komplexität ins System und müssen von Anfang an berücksichtigt werden, besonders wenn Änderungen in mehreren Fenstern mitverfolgt werden sollen, sind die Kopplungsfunktionen anspruchsvoll. Ist die Semantik einmal festgelegt muß darauf geachtet werden, die erforderlichen Informationen zur Zuordnung nicht zu verlieren.

Manche Datensätze sind zu groß/gewaltig um intensive Zuordnung zu erlauben.

Vorhandene Beziehungen zwischen Ansichten müssen aufgezeigt werden, wobei Fehlinterpretationen der Hinweise vermieden werden müssen. Ereignisse die in kurzen zeitlichen Abständen (100 Millisekunden) auftreten, werden als kausal verknüpft angesehen. Änderungen aufgrund von Aktionen in einer anderen Ansicht sollten daher diese Zeitspanne nicht überschreiten, da sonst der Zusammenhang nicht erkannt wird. Dauert die Aktualisierung einer Ansicht länger, sollte dieser Umstand (bzw. die damit verbundene Entkopplung) angezeigt werden

### **Einheitlichkeit**

*Gestalte die Benutzeroberflächen für MVs und deren Beschaffenheit einheitlich.*

Der Ausgleich für die Komplexität erfolgt durch die Leichtigkeit des Lernens, welches durch Einheitlichkeit unterstützt wird. Einheitliche Beschaffenheit erleichtert den Vergleich, im Gegensatz dazu kann Uneinheitlichkeit zu falschen Verknüpfungen beim Nutzer führen.

Neben den üblichen Empfehlungen zur Gestaltung von Benutzeroberflächen ist besonders auf die gleichbleibende System-Beschaffenheit und die Benutzeroberflächen-Möglichkeiten Wert zu legen.

System-Beschaffenheit umfasst den Datensatz und den Gesichtspunkt des Nutzers.

Beispiel - Palm - springt je nach Art des Kalenders nicht ins gleiche Menü

### **Aufmerksamkeits-Management**

*Nutze wahrnehmungs-Techniken um die Aufmerksamkeit des Anwenders zur richtigen Zeit auf den richtigen Bildausschnitt zu lenken.*

Treten Ereignisse auf, welche die Aufmerksamkeit des Nutzers erfordern, so kann diese leicht durch Animation, Töne, Hervorheben und Bewegung erregt werden.

Wenn der Anwender sich auf das System verlassen kann, muß er weniger oft die restlichen Fenster kontrollieren.

### **Schlußfolgerung**

*Tabelle mit den Richtlinien, in der die positiven und negativen Auswirkungen nochmals zusammengefaßt wurden. Es wurde die jeweils überwiegende Auswirkung eingetragen.*