Grafische Datenverarbeitung und Visualisierung

Dynamic Queries - Steuerelemente

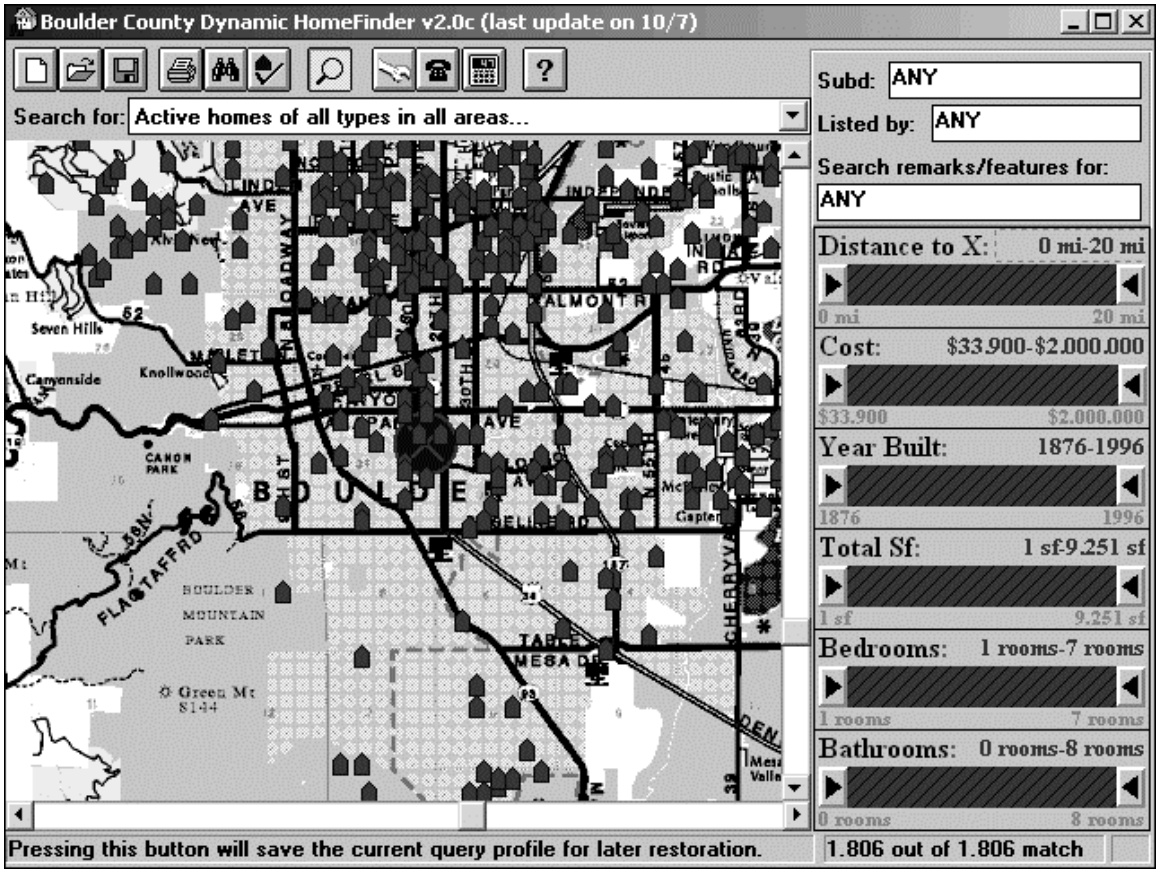
Gliederung der Präsentation

- Beispiel: „*Dynamic HomeFinder*“
- Problem: klassische Steuerelemente nicht geeignet für *Dynamic Queries*, insbesondere fehlt „Browsen“
- Lösung 1: erweiterte Steuerelemente
- Lösung 2: „Magic Lenses“
- Zusammenfassung und Ausblick

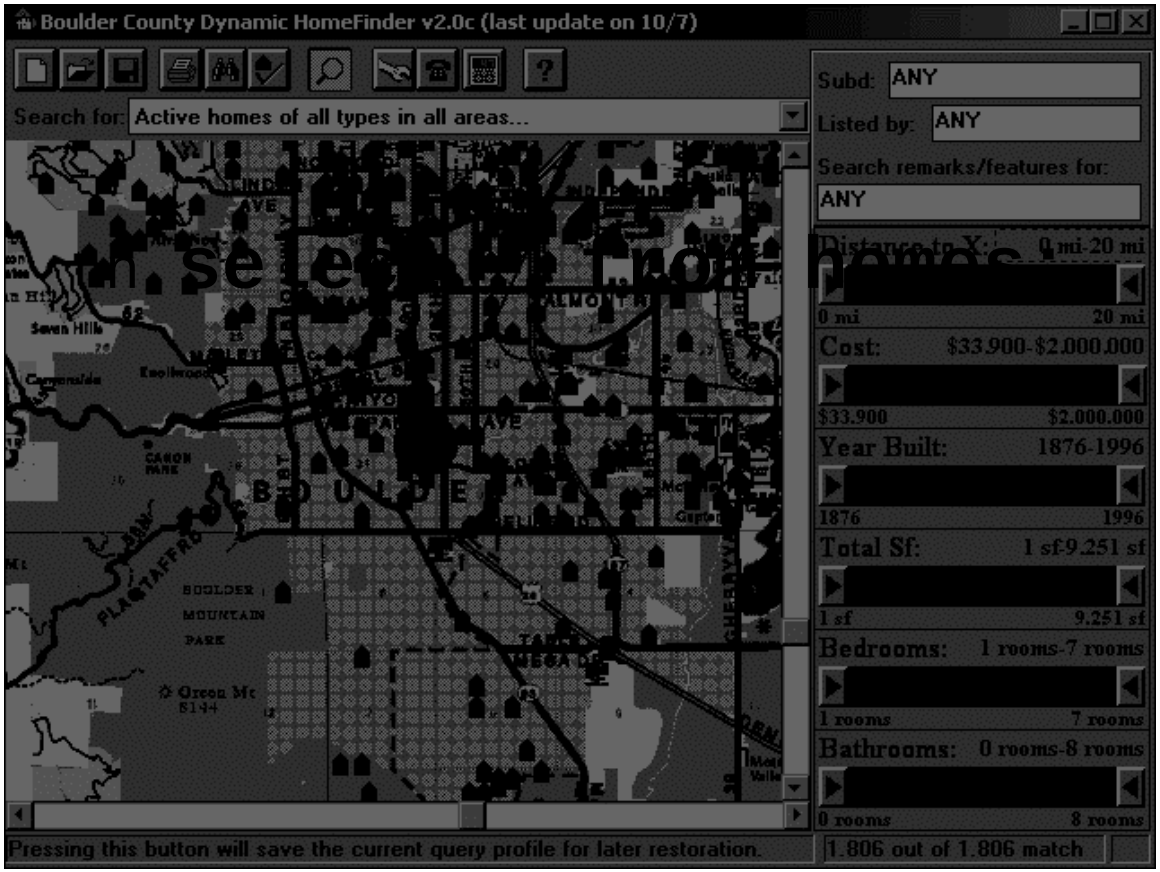
Beispiel: Dynamic HomeFinder

- Idee von Ben Shneiderman
- Die verwendete Variante wurde unter http://www.dqsoft.com/homefind/ zum Download gefunden, und von Christopher Williamson und Tom Smallwood entwickelt

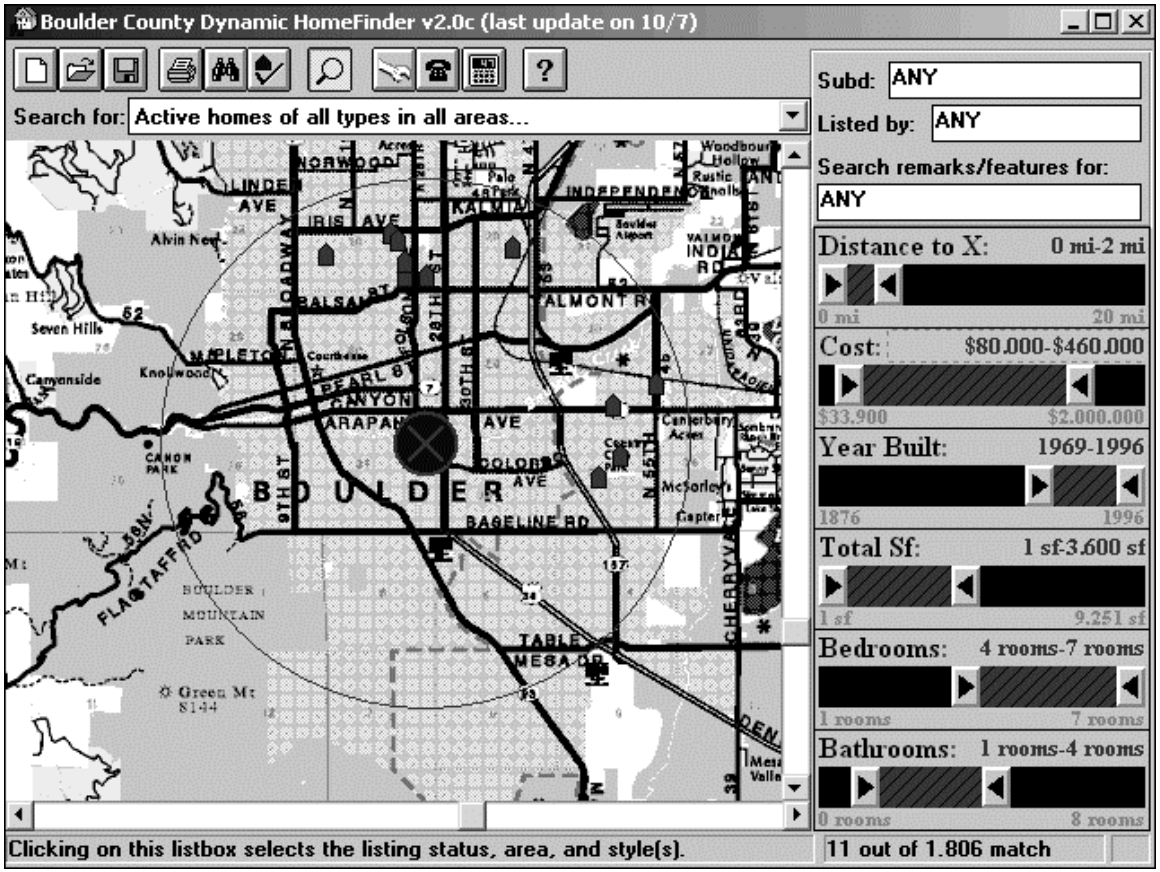
Beispiel:
Dynamic HomeFinder



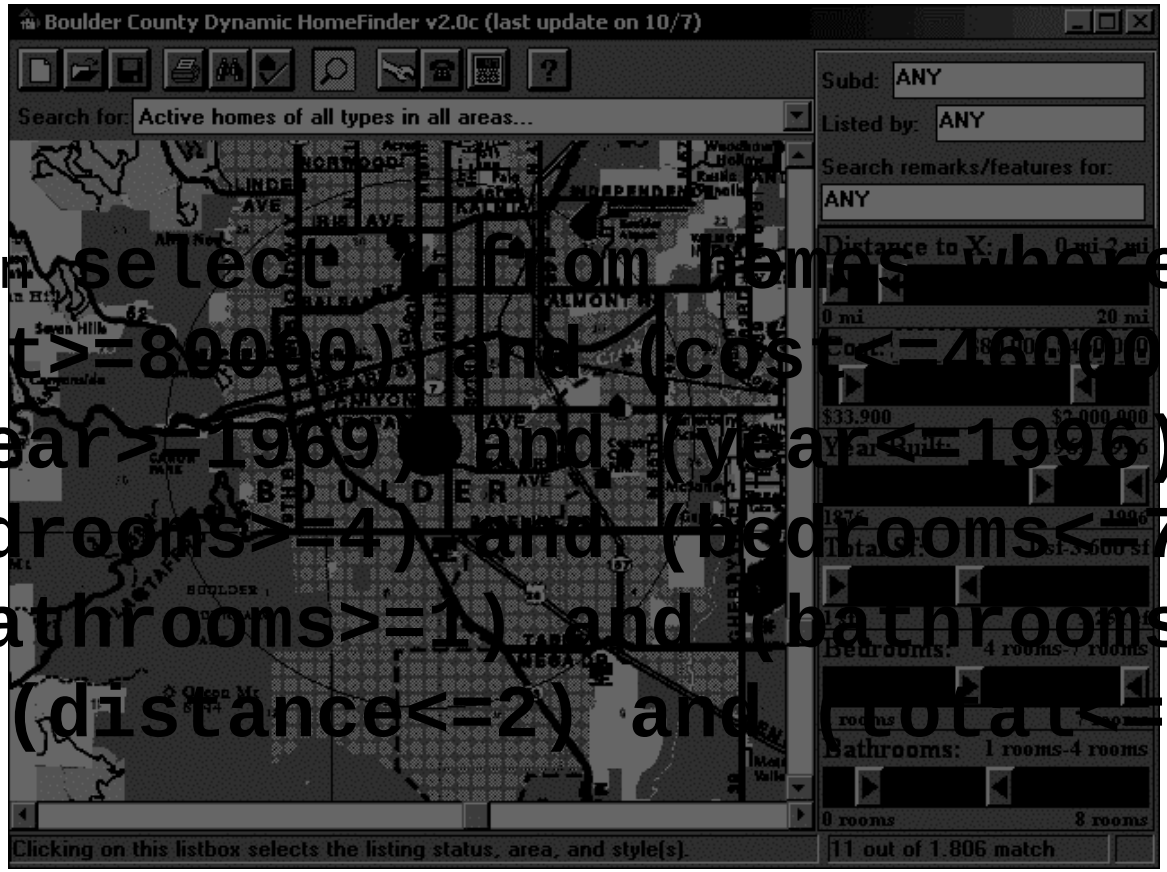
Beispiel:
Dynamic HomeFinder



Beispiel:
Dynamic HomeFinder



Beispiel: Dynamic HomeFinder




```
select * from homes where
(cost>=80000) and (cost<=400000) and
(year>=1969) and (year<=1996) and
(bedrooms>=4) and (bedrooms<=7) and
(bathrooms>=1) and (bathrooms<=4)
and (distance<=2) and (total<=3600);
```

Problem: Klassische Steuerelemente

- Slider
- Radiobutton
- Checkbox
- Combobox

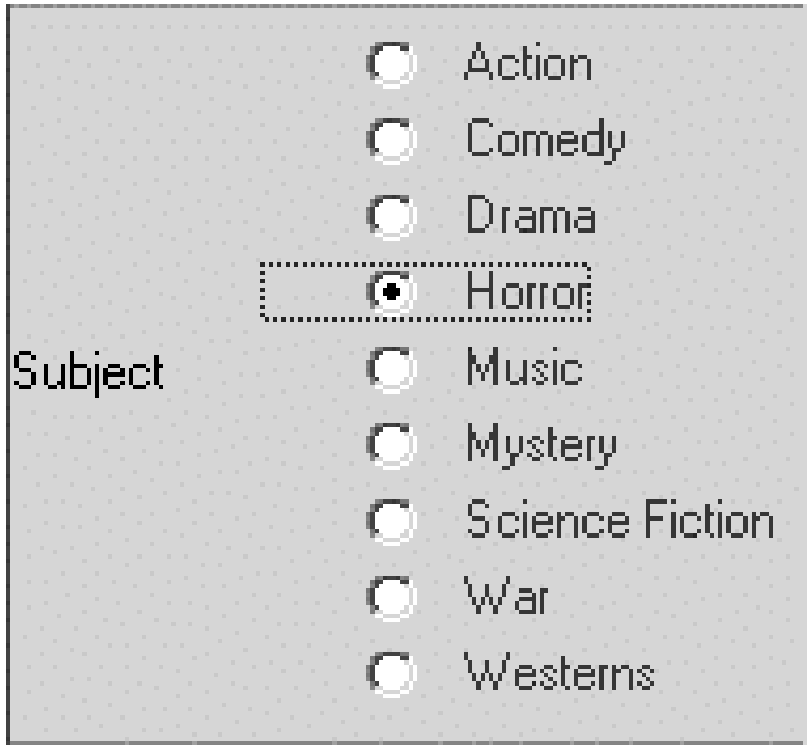
- Existieren unter Windows, KDE, Motif, OPEN LOOK, Swing, Mac OS, ...

Problem: Klassische Steuerelemente

- Slider (Schieberegler)
- Radiobutton (Optionsfeld)
- Checkbox (Kontrollkästchen)
- Combobox (Kombinationsfeld)

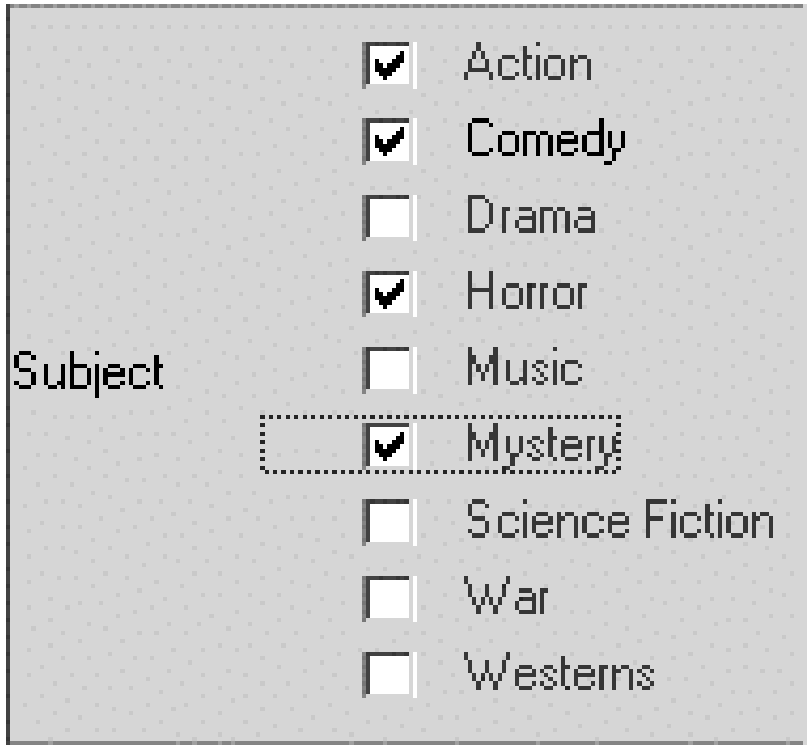
- Existieren unter Windows, KDE, Motif, OPEN LOOK, Swing, Mac OS, ...

Problem: Radiobutton (Optionsfeld)



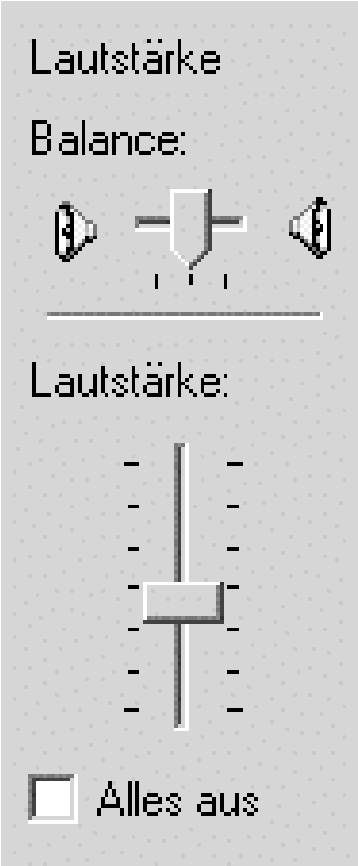
- Auswählen genau eines Wertes
- C Diskrete Werte gut auswählbar
- D Hoher Platzbedarf

Problem: Checkbox (Kontrollkästchen)



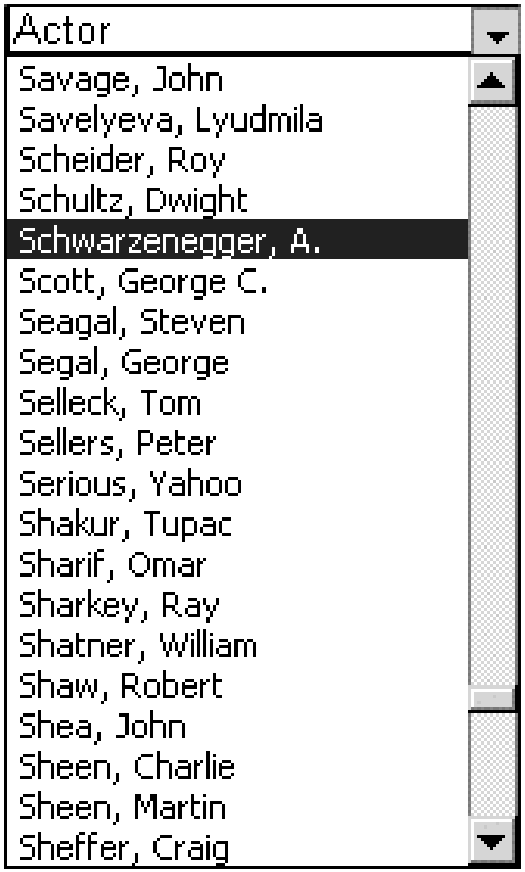
- Auswählen mehrerer Werte
- C Diskrete Werte gut auswählbar
- D Hoher Platzbedarf

Problem: Slider (Schieberegler)



- Auswählen genau eines Wertes
- D Diskrete Werte schlecht auswählbar
- C Konstanter Platzbedarf

Problem: Combobox (Kombinationsfeld)



- n Auswählen genau eines Wertes
- n C Diskrete Werte gut auswählbar
- n C Konstanter Platzbedarf
- n D Unkomfortable Bedienung

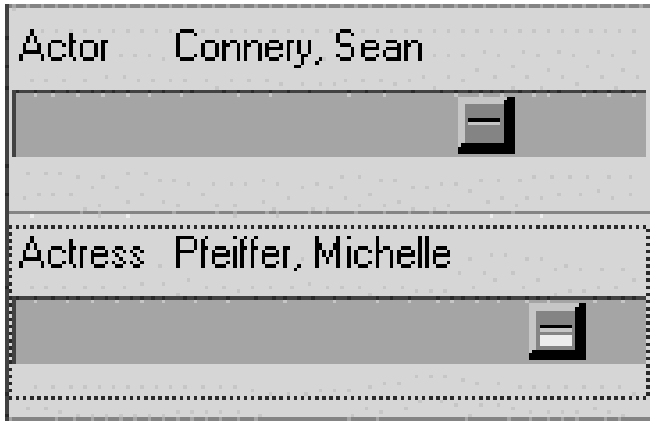
Problem: Zusammenfassung

	Diskrete Werte	Konstanter Platzbedarf	„Browsen“	Mehrfach selektion
Radiobutton	ý			
Checkbox	ý			ý
Slider		ý	ý	
Combobox	ý	ý		

Lösung 1: erweiterte Steuerelemente

- Alphaslider
- Data Visualization Slider
- 2D Widget
- in keiner gängigen Benutzeroberfläche integriert (noch?)
- Prototypcharakter

Lösung 1: Alphaslider



- Idee von Ben Shneiderman und Christopher Ahlberg
- Weiterentwicklung des Slider
- Auswahl einer textuellen statt numerischen Größe
- Auswählen eines Wertes oder Wertebereiches
- Diskrete Werte schlecht auswählbar
- „Browsen“ wird möglich, nötig für „tight coupling“

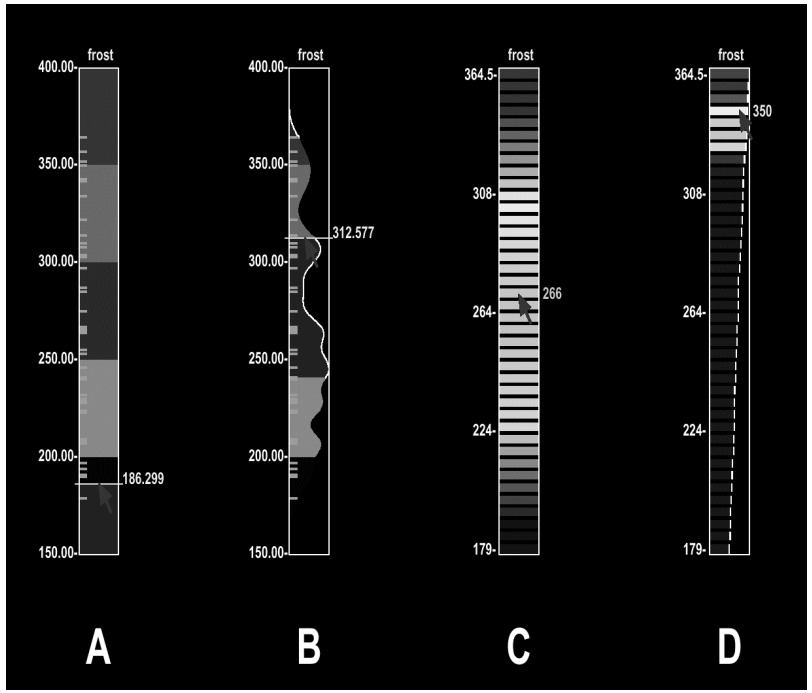
Lösung 1: Alphaslider

- Bereichsmarkierung
- Unscharfe Auswahl

Lösung 1: Zusammenfassung

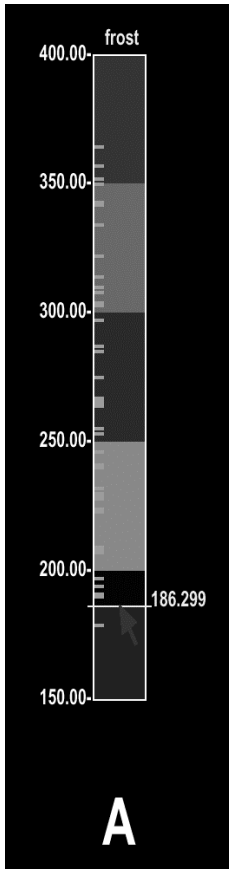
	Diskrete Werte	Konstanter Platzbedarf	„Browsen“	Mehrfach selektion
Radiobutton	ý			
Checkbox	ý			ý
Slider		ý	ý	
Combobox	ý	ý		
Alphaslider		ý	ý	

Data Visualization Slider



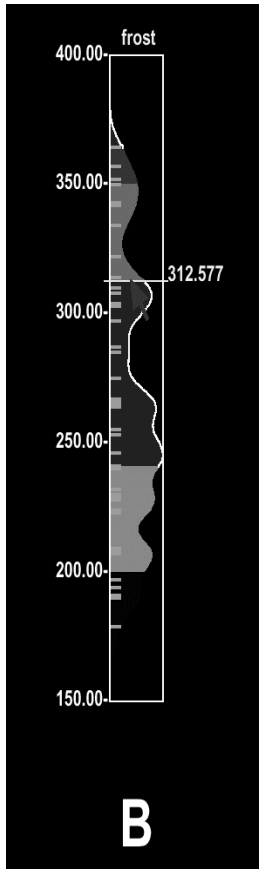
- Idee von Steven G. Eick
- Bereichsmarkierung
- Mehrfachselektion
- „Browsen“
- 4 Darstellungsarten

Lösung 1:
Data Visualization Slider



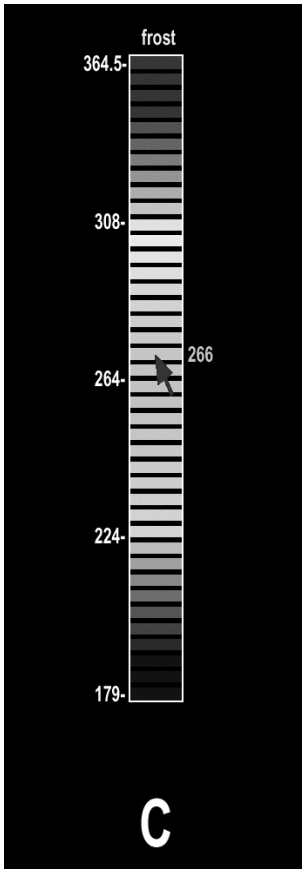
- Bereichsmarkierung
- Mehrfachselektion
- Kontinuierliche Werte

Lösung 1:
Data Visualization Slider



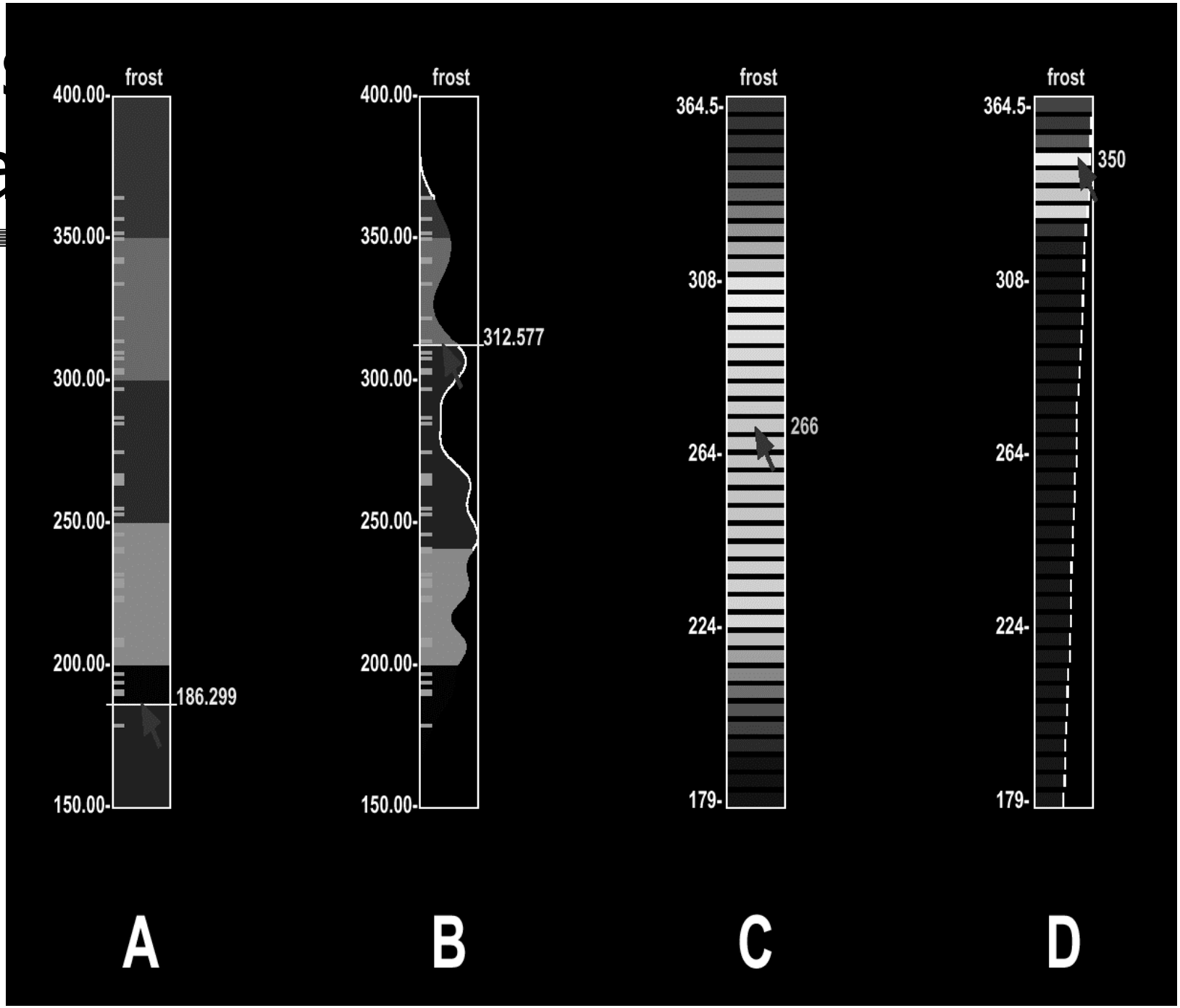
- Bereichsmarkierung
- Mehrfachselektion
- Kontinuierliche Werte
- Anzeige der Werteverteilung

Lösung 1:
Data Visualization Slider



- Bereichsmarkierung
- Mehrfachselektion
- Diskrete Werte

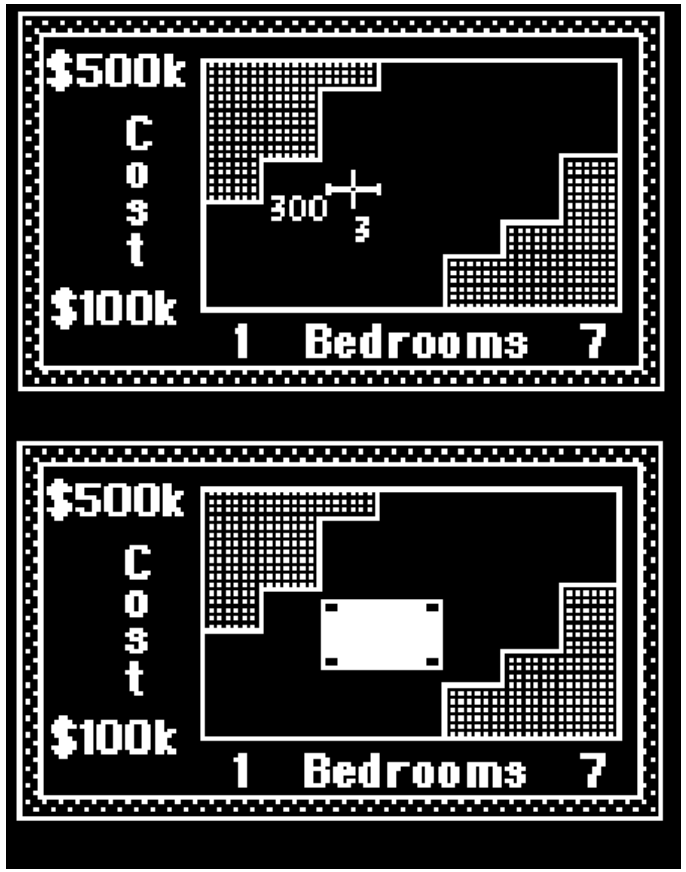
Lös
Da


frost
400.00
350.00
300.00
250.00
200.00
150.00
186.299
A
frost
400.00
350.00
300.00
250.00
200.00
150.00
312.577
B
frost
364.5
308
264
224
179
266
C
frost
364.5
308
264
224
179
350
D

Lösung 1: Zusammenfassung

	Diskrete Werte	Konstanter Platzbedarf	„Browsen“	Mehrfach selektion
Radiobutton	ý			
Checkbox	ý			ý
Slider		ý	ý	
Combobox	ý	ý		
Alphaslider		ý	ý	
D. V. Slider	(ý)	ý	ý	ý

Lösung 1: 2D Widget



- Idee von Ben Shneiderman
- zweidimensionaler „Slider“
- Bereichsmarkierung
- Mehrfachselektion

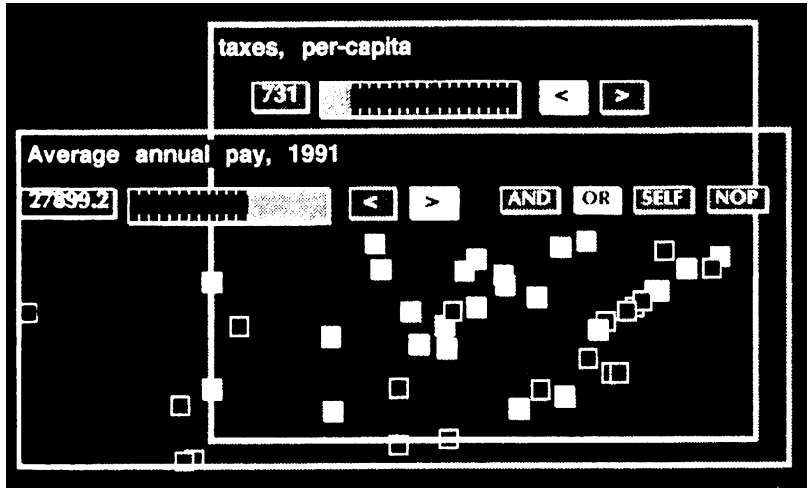
Lösung 1: Zusammenfassung

	Diskrete Werte	Konstanter Platzbedarf	„Browsen“	Mehrfach selektion
Radiobutton	ý			
Checkbox	ý			ý
Slider		ý	ý	
Combobox	ý	ý		
Alphaslider		ý	ý	
D. V. Slider	(ý)	ý	ý	ý
2D Widget		ý	ý	ý

Lösung 2: Magic Lenses

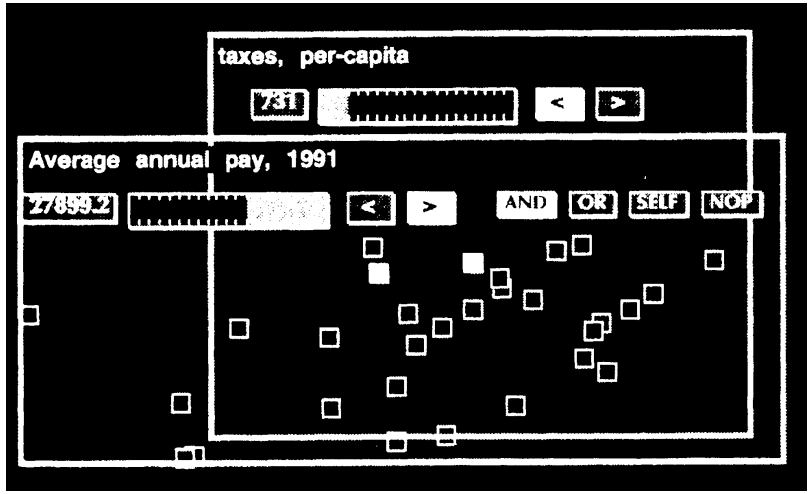
- Idee von Ken Fishkin
- Alle bisherigen Lösungen erlaubten keine Veroderungen von booleschen Ausdrücken

Lösung 2: Magic Lenses



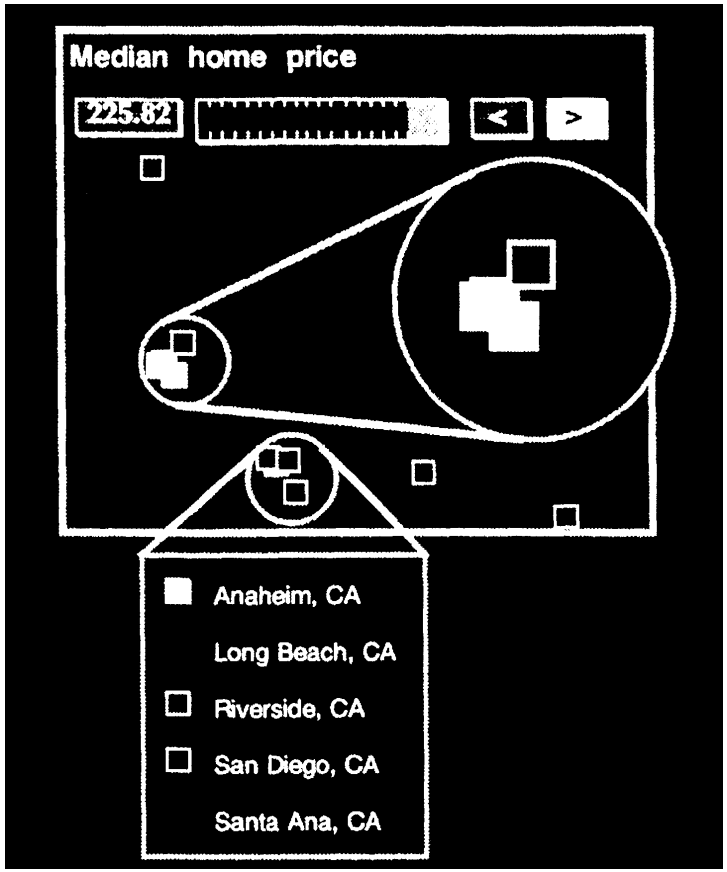
- Jede Linse kann als *UND*-, *ODER*-, oder *NOT*-Verknüpfung benutzt werden
- Linsen können gruppiert werden („Klammerung“)
- Linsen sind beliebig verschiebbar

Lösung 2: Magic Lenses



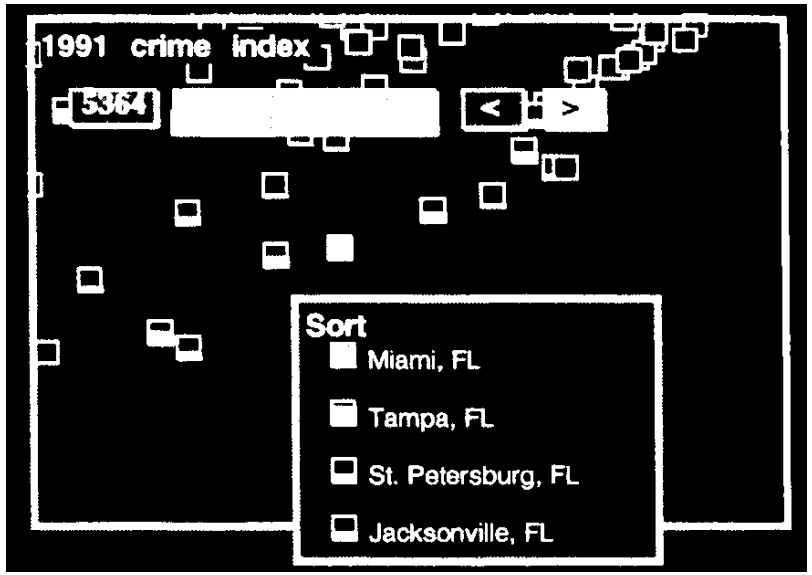
- Änderungen an den Einstellungen wirken sich in Echtzeit aus
- „ausgeschiedene“ Datensätze sind zu erkennen
- Multiple Views

Lösung 2: Magic Lenses



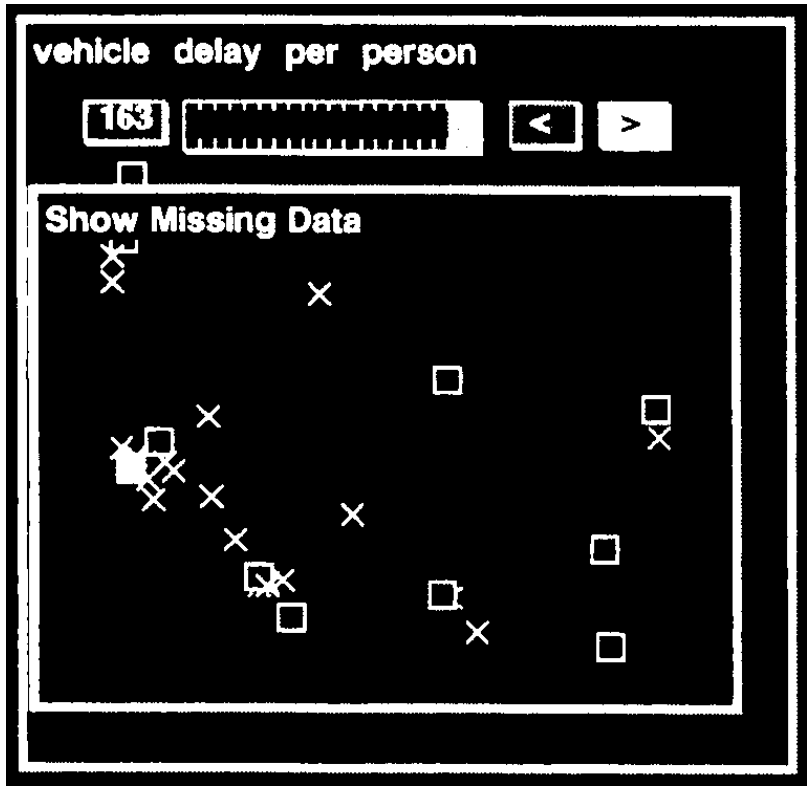
- Details on Demand

Lösung 2: Magic Lenses



- Unscharfe Werte („Fuzzy Logic“)

Lösung 2: Magic Lenses



- Missing Values in schwacht besetzten Datenfeldern sind klassisches Problem
- Sichtbarmachung durch „Missing Data Linse“

Zusammenfassung

- Dynamic Queries:
  „spielerisch“ Datenbankabfragen erzeugen
- Faszinierende Möglichkeiten ⇔ hohe Kosten:
  Gegenwärtige Hardware, Algorithmen, Datenstrukturen, Datenbanksysteme, Benutzeroberflächen und Protokolle genügen den neuen Anforderungen nicht!

Ausblick

- Bisher ist anwendungsspezifische Entwicklung nötig. Allgemeine Lösungen? Standardsoftware? Insbesondere fehlen Entwicklungs-werkzeuge.
- Probleme für Sehbehinderte (Audio Feedback?)

Ausblick

- Komplexe Datenbankabfragen? (group by, ...)
- Es gilt die Universalität der klassischen Abfragesprachen (SQL, ...) zu erreichen

Ende

- Autor: Fabian Wleklinski
- E-Mail: fabian@wleklinski.de