

NoteCards og 7+1 design issues

■ NoteCards

- Halasz, Moran & Trigg: Notecards in a Nutshell

■ 7 Issues:

- Halasz: Reflections on NoteCards: Seven issues for the Next Generation Hypermedia Systems

■ +1:

- Meyrowitz: The Missing Link: Why we're all doing hypertext wrong

Notecards baggrund

- Blev udviklet på Xerox PARC i midten af '80'erne
- Udviklet af Randy Trigg, Tom Moran og Frank Halasz
 - Frank og Randy var senere med i Dexter gruppen
 - Randy har senere arbejdet ved IMV og DAIMI i tre år, og har været med til DHM design
- Generelt hypermedia system til mange anvendelsesområder
 - Har været i udbredt brug i forskningsmiljøer
- Udviklet i Xerox Lisp (InterLisp), der kørte på dedikeret hardware
 - Xerox Lisp og NoteCards kan vist stadig købes fra et Xerox spin-off firma, der hedder Envos
- “Seven Issues” artiklen
 - er en meget refereret artikel pga. fremtidsvisionerne

*Vi har en gammel Xerox Lisp og NoteCards her på DAIMI ,
den kan gen-installeres, hvis nogen har arkæologiske interesser*

NoteCards begreber

■ Små notecards = nodes

- “kort” (vinduer) med indhold
- flere typer: tekst, grafik, bitmaps
- kan tilknyttes attributter
- standard Interlisp editorer benyttes (TEdit, Grapher, etc.)
-

■ Links

- point-to-node (fra indlejret link ikon til helt kort), to endepunkter
- to-vejs links, på hvert kort kan åbnes et property vindue, der viser link ikoner for alle ind og udgående links
- typer i form af labels
- link ikoner viser valgfrit: destinationstype, link label, destinationsnavn
- følges ved dobbeltklik på link ikoner

Link endepunkter - generel terminologi

- **node-to-node**
 - fra en knude til en anden knude - typisk i browsere
- **point-to-point**
 - fra icon i en knudes indhold til icon
- **point-to-node**
 - fra icon i en knudes indhold til en knude i sin helhed
- **span-to-span**
 - fra et tekst-område til et andet tekst-område
- **span-to-node**
 - fra et tekst-område til en knude i sin helhed
- **Osv.**

NoteCards begreber...

■ grafiske browsere

- en kort type med grafer, der viser (en del) af link netværket
- kan beregnes af systemet ud fra specifikation af links, der skal inkluderes
- brugeren kan bruge dem til at editere netværket
- implementeret vha. en 'browser' link type, repræsentationen af kort i browsere er link ikoner
- mini-oversigt kan bruges til navigation

■ hierarkier: fileboxes

- ethvert kort skal tilhøre mindst en filebox, ved sletning af sidste 'filecard' link flyttes kort til orphans boxen
- fileboxes er også implementeret med 'filecard' links
- et kort kan indgå i mange fileboxes
- bruges f.eks. til at generere lineært output

NoteCards begreber...

■ Søgning

- på keywords i property liste samt fuld-tekst søgning
- resultater af søgninger optræder som et kort med links til “hits”

■ Tailorability

- API med mere end 150 Lisp funktioner
- preferencer til at styre interaktionen
- der kan f.eks. bygges nye kort typer
- mange eksempler på udvidelser med specialiserede korttyper: GuidedTour, TableTop, Animation, Collaborator, History, ...

Seven Issues

- 1 **Search and Query**
- 2 **Composites**
- 3 **Virtual structures**
- 4 **Computation**
- 5 **Versioning**
- 6 **Collaboration**
- 7 **Extensibility and Tailorability**

Diskussionen baserer sig på tre års brugserfaring med NoteCards, og der fokuseres på principielle svagheder

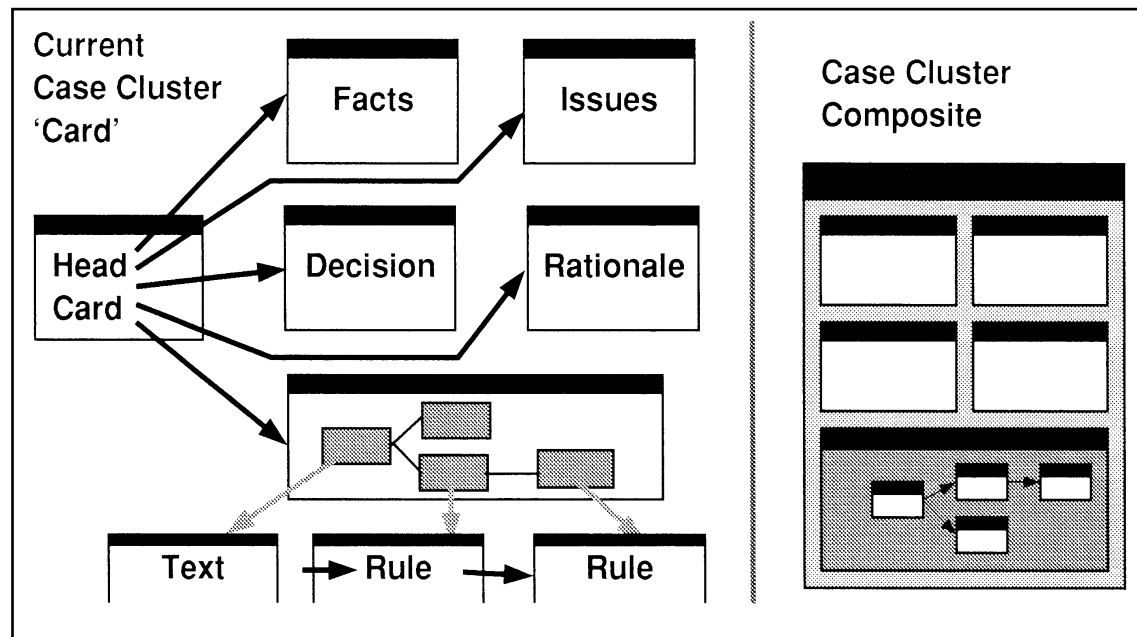
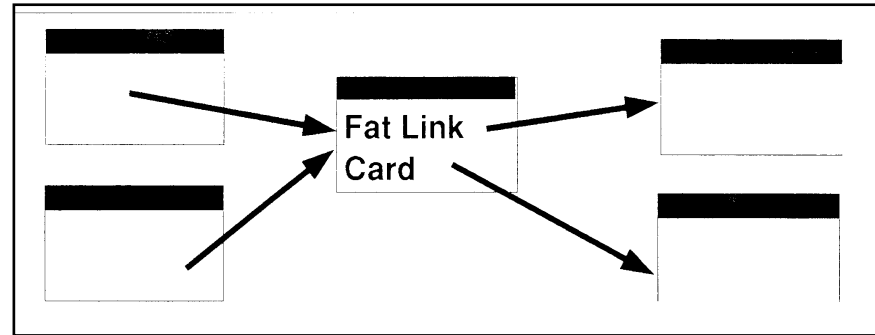
1. Search and Query

- **problemer med at navigere i store, ukendte og heterogene netværk**
 - selv 500 kort giver problemer
- **link following ikke optimal, når man “ved” hvad men leder efter**
 - f.eks. information om et bestemt afgrænset emne
- **behov for avanceret query mekanisme**
 - søgning efter indhold (f.eks. et ord i tekstknuder)
 - søgning efter struktur (f.eks. nodes med en indgående “support” link og en udgående “counter-argument” link eller transitive lukninger af bestemt link type)
- **ufordringer**
 - designe et query “sprog”, med regulære udtryk for grafer. Feks. query by example grafisk brugergrænseflade
 - lave en effektiv “søge maskine”, der kan håndtere specificerede queries
- **queries kan benyttes som filtre til generering af browser kort**

2. Composites

- **Behov for gruppevis håndtering af nodes og links**
 - eksempel CaseCluster fra en juridisk applikation,
 - 10 sammenlinkede kort som et skelet: 8 kort med indhold og 2 med meta information og struktur
 - **PROBLEM:** et CaseCluster kan ikke repræsenteres i sin helhed i browsere
- **Problem: Filebox begrebet blander inclusion og reference**
 - metaforen antyder inclusion, men implementeres som reference
 - inclusion bør understøttes i et hvert hypermedia system
 - det er af sekundær betydning, om der benyttes links til implementation bare semantiken er klar
- **Problem: document card indholder kopi af inkluderede kort istedet for at være et dynamisk aggregat af disse**
- **FORSLAG: Composite nodes - første classes nodes, som kan "indeholde" andre nodes og links.**

Behov for composites (Halasz, 1990)



2. Composites (design spørgsmål)

- Kan nodes indgå i mere end én composite?
- Kan links udpege nodes indeni composite nodes? Hvordan følges sådanne links?
- Hvordan håndteres versionering? Skal ny version af indre node medføre ny version af composite?
- Skal composites implementeres med links eller med en ny konstruktion?

3. Virtual structures

- **Problem: håndtering af dynamisk information**
 - hypermedia model er for statisk
 - netværket kan ikke opdatere sig selv, når en del af indholdet ændres
- **Problem: “premature organization” - brugeren forventes at kende struktur på forhånd**
 - organiseringen halter bag efter erkendelsen
- **Behov: virtuelle strukturer - baseret på intentionel specifikation istedet for extensionel som links**
 - eksempel: composites, der genberegner deres indhold ud fra en query
 - runtime strukturer, men skal optræde overfor brugeren som almindelige nodes
 - browsere som virtuelle composites, men der skal kunne tilføjes statisk definerede links mv.
- **Virtuelle links: links hvor destination er specificeret intentionel**

4. Computation

- **PROBLEM:** hypermedia model er passiv
 - ingen støtte for automatisk skabelse eller behandling af nodes og links
- **Forslag:** Indarbejdelse af ekspert system teknologi til at processere hypermedia strukturer
 - eksempel med script cards edb-støttet instruktion
- **Design valg:**
 - ekstern beregningsmaskine, der bruger API, eller
 - indbygget beregningsmaskine

5. Versioning

- **Behov for at følge historikken i netværkets udvikling**
- **Eksisterende løsninger versionstråd for individuelle nodes og links**
 - e.g. KMS, Neptune,...
- **Forslag: versioner som virtuelle enheder**
 - både versioner og deltaer som nodes
 - dvs. både versioner og deltaer kan annoteres
- **Medfører flere semantiske fortolkninger af referencer til objekter**
 - specifik version, seneste version, seneste version i bestemt deltræ
- **Alternativ: layer/context begreb**
 - koordineret sæt af ændringer - et release

6. Collaboration

- **Hypermedia er velegnet til at støtte samarbejde**
 - annotering/kommentering, multiple views, asynkron bidrag til fælles skriveprocesser
- **Problem: NoteCards var et enkeltbrugersystem**
 - samarbejde foregik udenfor systemet
- **Behov for simultan adgang til hypermedia struktur**
 - Ingen systemer tilbyder tilstrækkelig avanceret concurrency kontrol
 - Ingen tilbyder beskeder til brugere, når andre ændrer i strukturen
 - Ingen støtte til social interaction
- **Forslag**
 - Langtidstransaktioner
 - “Soft locking”: f.eks. linke mens andre skriver
 - “Notification on critical events”
 - Støtte til social interaction: substantive, annotative, og procedural
 - Hypermedia retorik til at øge fælles forståelse

7. Extensibility and Tailorability

- **Behov for at kunne tilpasse generelt hypermedia system til bestemte anvendelsesområder**
- **Nuværende løsninger: API**
 - **problem: kun programmører kan udføre tilpasninger**
- **Forslag: studere mulighederne for at give ikke-programmører adgang til tilpasning**
 - **“substrate” lag og fortolket sprog + direkte manipulation**

Issue 7+1 (Meyrowitz)

- **Observation: hypermedia er ikke en udbredt kommerciel success**
 - hypermedia er absolut *ikke* tilstede på enhver desktop (i 88-89)
- **Problem: Eksisterende systemer er ø'er eller monoliter**
 - brugere skal opgive alle deres kendte programmer og flytte til speciel hypermedia verden
- **Forslag: Implementer linking på operativsystem og toolbox niveau ligesom cut 'n paste**
 - denne løsning er valgt på GO's platform til Palmtops