

# 19. Information Organization in/for User Interfaces

---

INFO 202 - 3 November 2008

Bob Glushko

## Plan for Today's Lecture

---

Broad and narrow definitions of Information Architecture

Information Visualization

Principles for Information Architecture

- Separation of content from structure and presentation
- Structuring principles
- Reinforcing structure with presentation
- Internationalization and localization

User interface design patterns

# Defining "Information Architecture"

---

An architecture describes a system's components (or "building blocks") and their relationships with each other, using hierarchical and compositional structure to define the component boundaries

This gives an "architected" system an explicit model, in contrast with systems that are implemented incrementally without a master plan or without the effort to create reusable abstractions and components

Abstract patterns of information content or structure are sometimes called architectures

So we define "Information Architecture" by combining the definitions of these concepts:

- "IA is designing an abstract and effective organization of information and then exposing that organization to facilitate navigation and information use"

---

## A Formula for Information Architecture

---

**INFORMATION ARCHITECTURE =  
((CONTENT MODEL +  
INFORMATION STRUCTURE) +  
NAVIGATION STRUCTURE)**

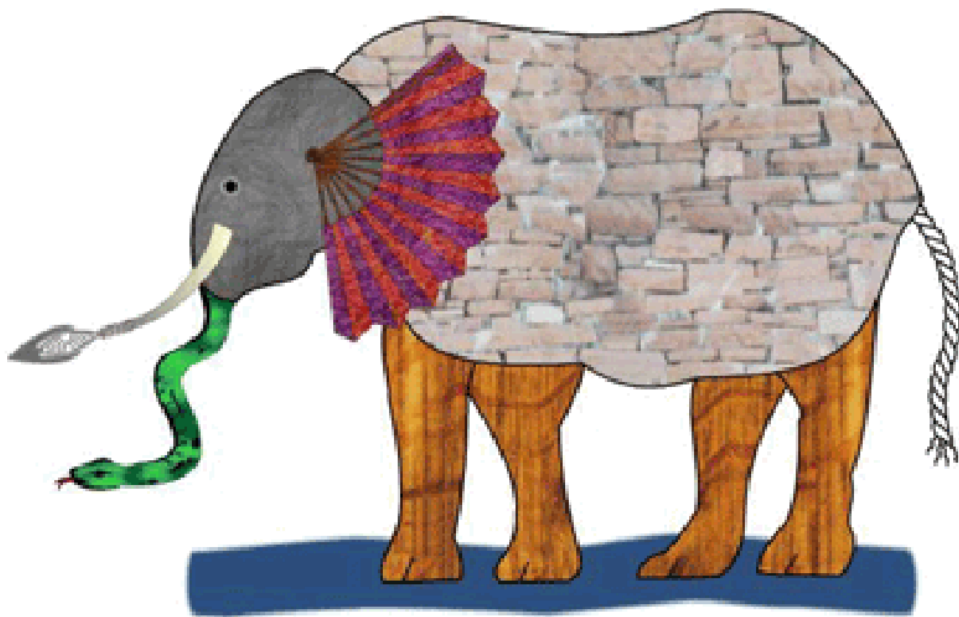
# A Formula for Information Architecture in User Interface Design

---

**USER INTERFACE DESIGN =  
(*Information Architecture* +  
INTERACTION DESIGN) +  
PRESENTATION DESIGN)**

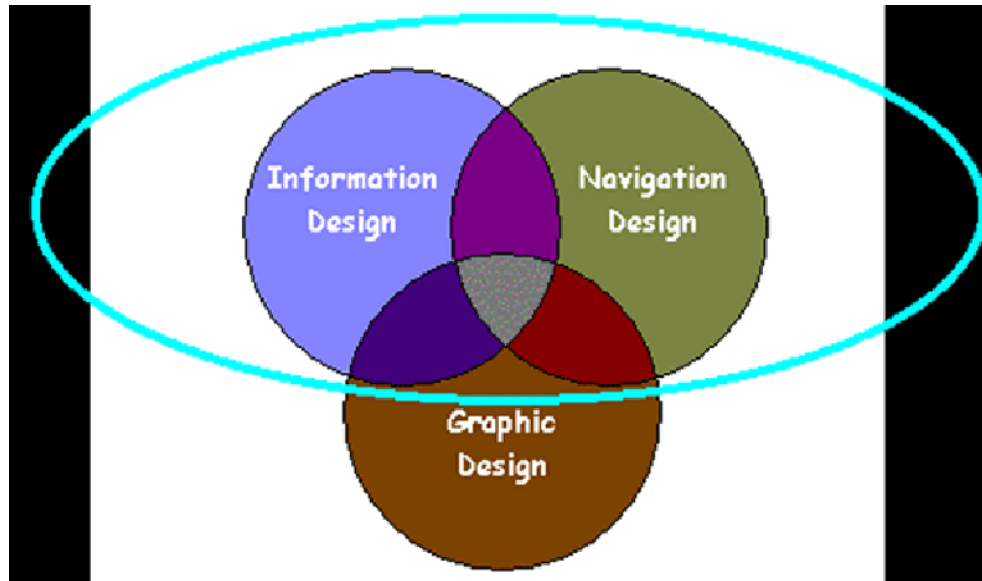
**But Not Everyone Agrees with this Definition  
for IA**

---



# A More Typical Depiction of IA

---



IA defined by the ellipse, which brings in graphic design and doesn't establish content modeling as the foundation

## Bounding the Scope of Information Architecture

---

Most people would include:

- Content organization / data modeling
- Navigation / interaction design

Some people would also include:

- Visual / graphical design
- Information visualization

Some even include:

- "Experience" design
- "Virtual environment" / "mediated realities" design

# Information Architecture as the Foundation for User Interface Design

---

If you limit the scope of IA to conceptual and structural modeling, IA can follow robust and consistent principles

UIs designed on an IA foundation likewise can follow robust and consistent principles

Many UI design and implementation activities can be automated or "model-based"

There can be substantial reuse of design patterns or components

## If User Interface Design isn't (or can't be) based on IA

---

UI design follows less robust and more iterative methods

- Heuristic evaluation
- Prototyping
- Usability testing to revise prototype (incremental addition of functions, features, and "business rules")

# Nielsen's Ten Usability Heuristics

---

A classic set of usability heuristics is Jakob Nielsen's [http://www.useit.com/papers/heuristic/heuristic\\_list.htm](http://www.useit.com/papers/heuristic/heuristic_list.htm)

Note how this iterative methodology interleaves or revisits IA design activities with UI design ones

- "The system should speak the users' language, with words, phrases and concepts familiar to the user"
- "Follow real-world conventions, making information appear in a natural and logical order"
- "Follow platform conventions"
- "Dialogues should not contain information which is irrelevant or rarely needed"

---

## Information Visualization

---

Information visualization is the depiction of information using spatial and graphical conventions

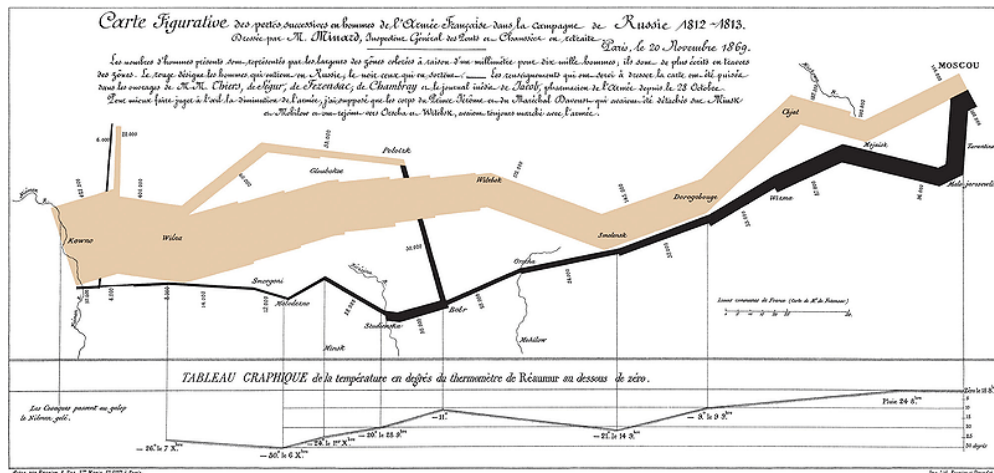
IV most often involves computer-based, interactive visual representations to exploit human perceptual and cognitive capabilities for perceiving and understanding more information than can be obtained from textual displays

Two primary goals for IV:

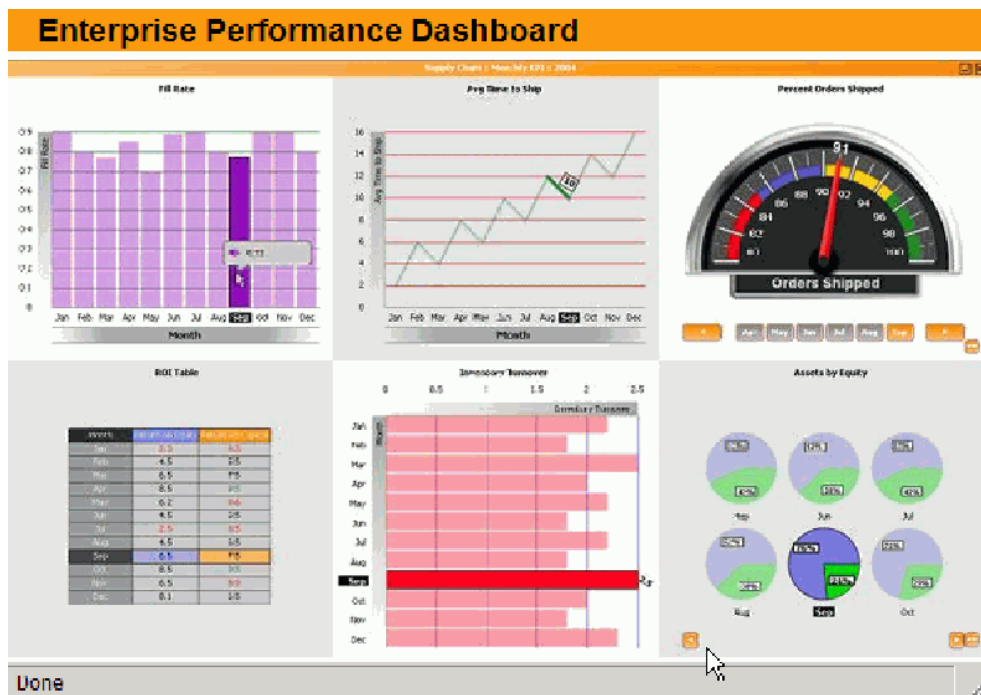
- Information Producer Goals: To explain, illustrate, and communicate
- Information Consumer Goals: To facilitate analysis, exploration, and discovery

IV is especially useful for large multidimensional data sets where "sensemaking" requires taking multiple points of view at different levels of abstraction

# Minard's Famous Information Graphic



# Information Visualization for Business Intelligence



# Info Vis vs. IA vs. UI

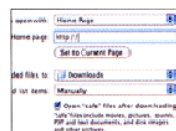
Information Visualization as a design domain has a unique relationship to both IA and UI design as we've defined them today

Like IA, Info Vis pays careful attention to content models and structural relationships

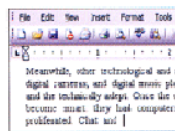
Like UI design, Info Vis seeks to apply interaction structure and presentations to the underlying information

But Info Vis often succeeds by transcending or ignoring the underlying information models and instead imposing creative alternatives to them

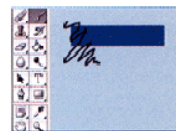
## User Interface Design Idioms (Tidwell)



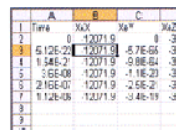
Forms



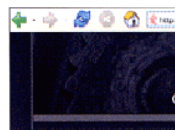
Text editors



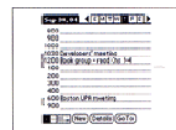
Graphic editors



Spreadsheets



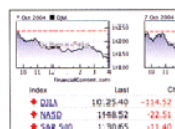
Browsers



Calendars



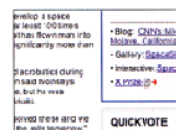
Media players



Information graphics



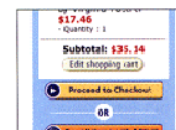
Immersive games



Web pages



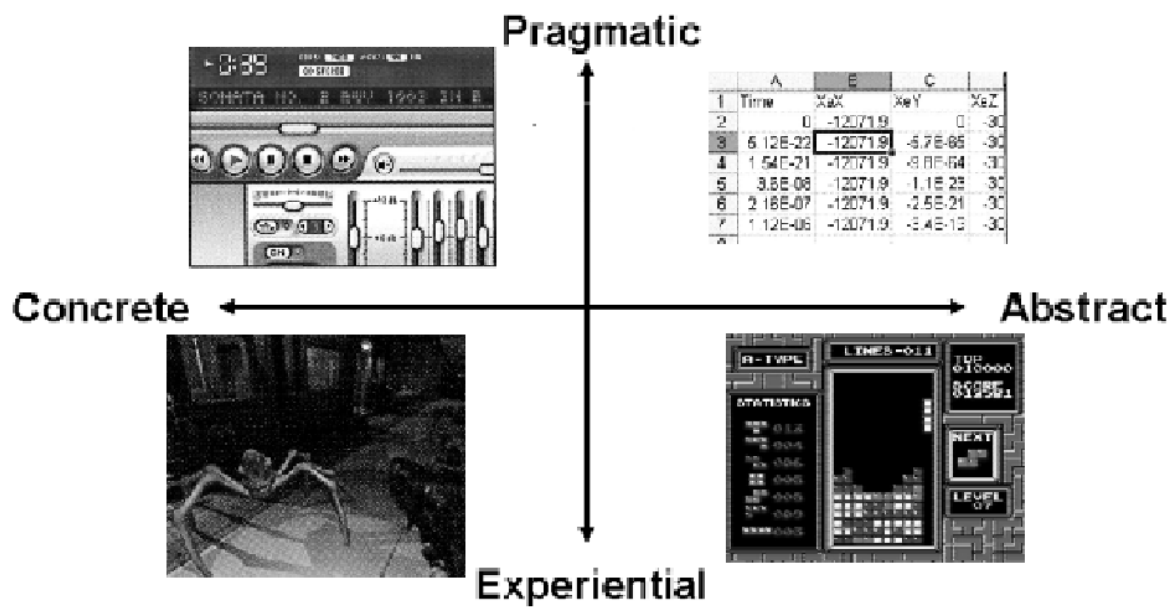
Social spaces



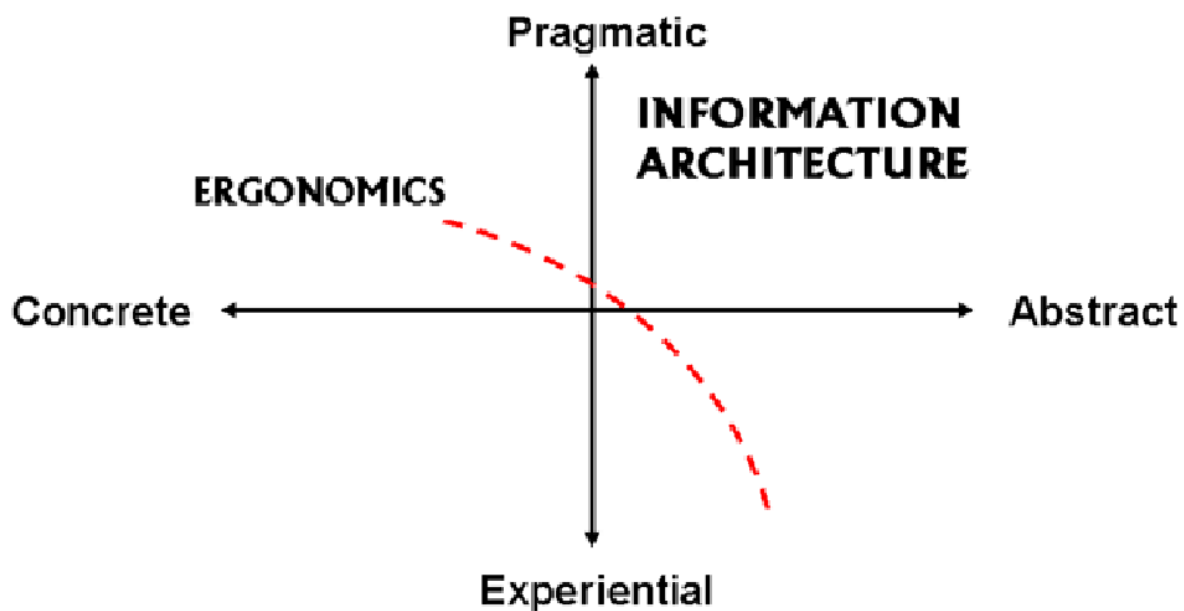
E-commerce sites



# Where is IA most Critical to UI design?



# Narrowing the Scope of IA



# Application Types in this Narrow Scope

---

Applications in this region of the overall design space involve information that is explicitly organized -- governed by rules or constraints

The applications can be thought of as software artifacts that present, collect, and manipulate information according to these rules or constraints

For "Transactional" applications the rules are encoded in document type and process models from some specific domain

Many "Visualization" applications operate on document and data models from multiple domains and apply additional structural and presentational models (or "metaphors") to them

---

## Model-based Transactional IA [1]

---

Generate the application (or a "scaffold" for it) from the models

The model can be directly used to generate the software needed to manipulate, transform, or display instances

The model can be interpreted by a generic software platform to configure its behavior

(We'll talk more about model-based user interfaces in the next lecture)

## Model-based Transactional IA [2]

---

Evaluate the model-based application with usability or other \*-ility techniques  
Revise the models as suggested by these evaluations  
Regenerate the application from the revised models

## The Most Important Principle for Information Architecture

---

We say "the document is about ... the photograph is about... the movie is about"

We're expressing a distinction between information as conceptual or as content: and the physical container or medium, format, or technology in which the information is conveyed

It is very useful to think abstractly about "information content" without making any assumptions or statements about the "presentation" or "rendition" or "implementation"

Separating content from its structure and presentation is the most important principle of Information Architecture

# Three Types of "Stuff" or Kinds of Information

---

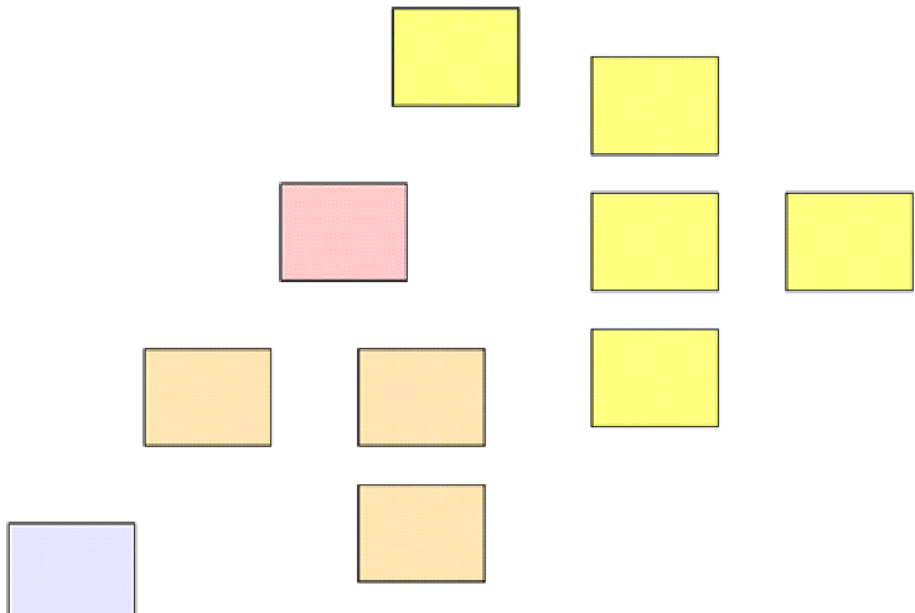
*Content* – "what does it mean" information

*Structure* – "where is it" or "how it is organized or assembled" information

*Presentation* – "how does it look" or "how is it displayed" information

## Content Components

---



# Structural Information

---

Physical piece of a document or user interface (e.g. table, section, header, footer, panel, window)

Embodies the rules on how content components fit together, often hierarchical

Often driven by context of document use

Most applications and web sites are organized with a small set of structures:

- Lists/hierarchies
- Networks/links

# Applying Structure

---

The structural components provide the hierarchical "skeleton" or "scaffold" into which the content components are arranged; the structure remains fixed when the content changes

Structural components are often identified by the names attached to pieces of information – think of the outline or table of contents or lists of various kinds

Frequently a close relationship between structural and presentation items, especially in a paper document. This goes some way to explaining why the document-centric school places such strong emphasis on structural components.

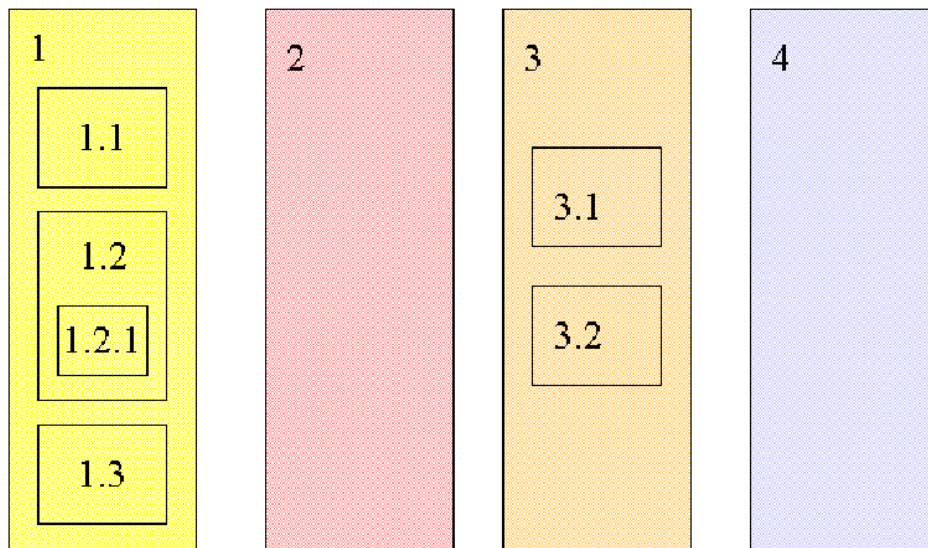
# Structure is Independent of Content

---



# Structural Relationships Among Components Expressed as a Hierarchy

---



# Lists

---

Common types of lists in user interfaces and applications:

- Lists of objects -- e.g., an inbox full of email messages
- Lists of actions or tasks -- e.g., browse, buy, sell, register
- Lists of subject categories (or facets) -- e.g, health, science, technology
- Lists of tools -- e.g., calendar, address book, notepad

# Entry Points

---

Similar to list structures are "entry points" -- structures that are "wrapped around" some set of content components to provide an organized way to access them

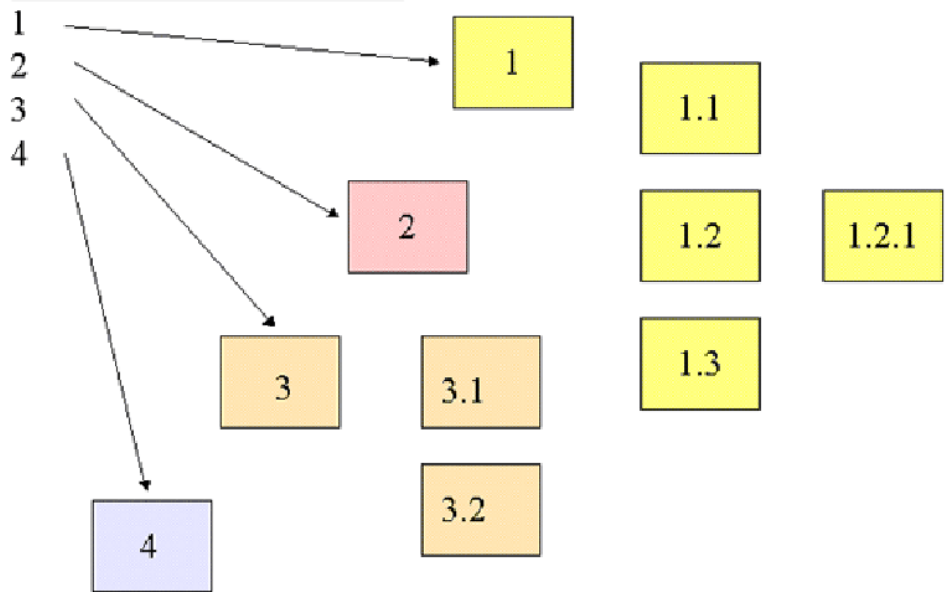
Most familiar examples are tables of contents and topical indexes; these are created from the names or other descriptive metadata for each component (which might first be extracted by processing the component content)

An entry point can be created as a static structure at design time, but preferably would be dynamically generated at run time

There are many similar examples of entry point structures generated from the names or descriptors of content components (Tables or Lists of content of type "X")

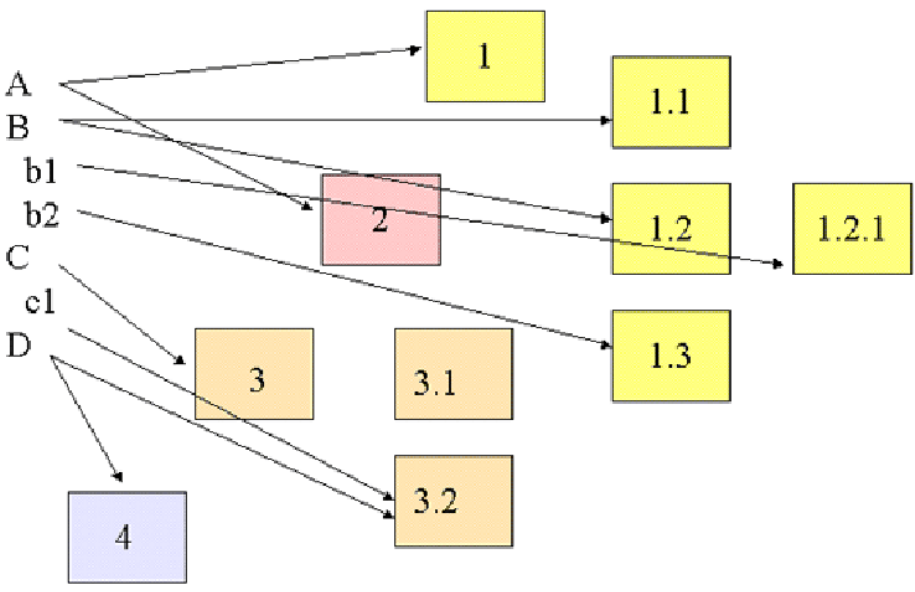
# Table of Contents as "Entry Point"

## TABLE OF CONTENTS



# Topical Index as "Entry Point"

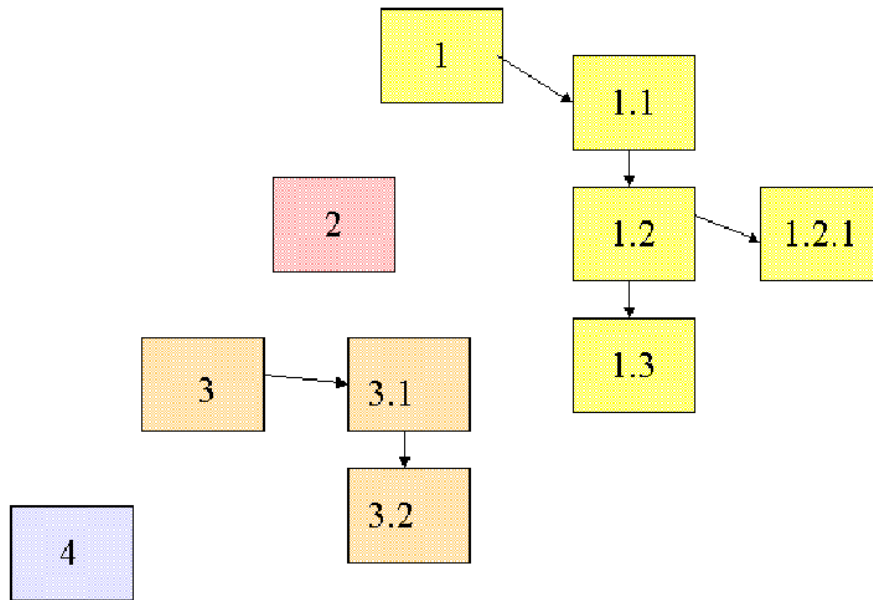
## TOPICAL INDEX





# Structural Relationships Among Components Expressed as a Network

---



## Links

---

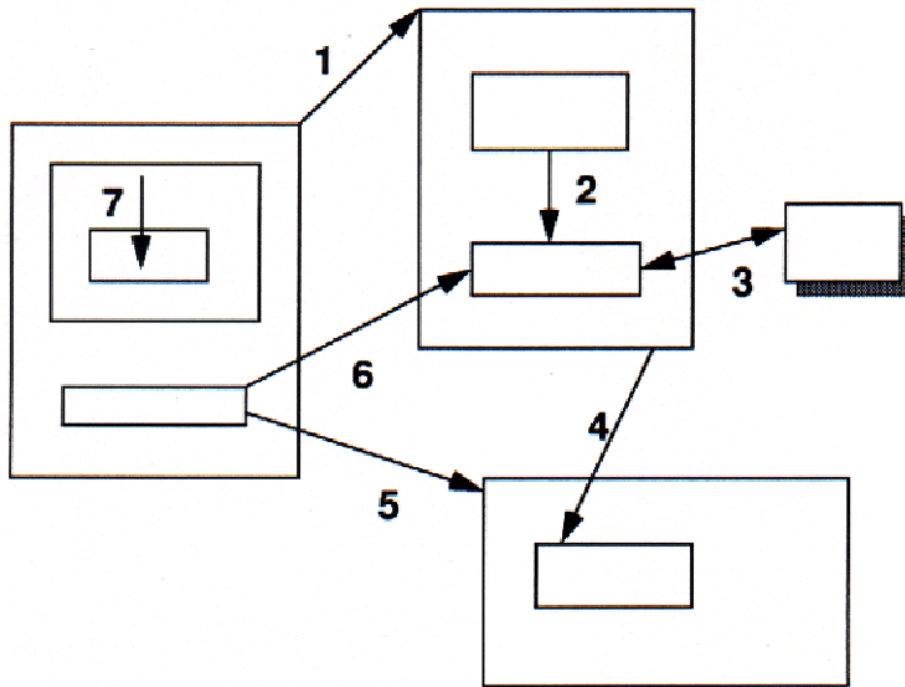
Links are relationships between components that can express content as well as structural information

A link is represented in a logical model by its:

- Anchors -- the point, region, or span within the components to which it refers
- Type -- the semantics that the link relationship represents; not always explicit
- Directionality -- is the link one or two-way? Is the relationship meaningful in both directions? Does the reverse direction link mean the inverse?
- Cardinality -- 1 to 1 to many?

# Link Structures

---



# Navigation Structures

---

Navigation structures support finding or moving between components

- Forward or back in some structural organization
- Forward or back in a temporal organization (history list) or according to other relationships associated with the content components

# Presentation Information

---

Human-oriented attributes for visual (or other sensory) differentiation (type font, type size, color, background, indentation, pitch, ...)

Good user interface design correlates this with structural or content information

Presentation affects structure and content by applying transformation rules to them

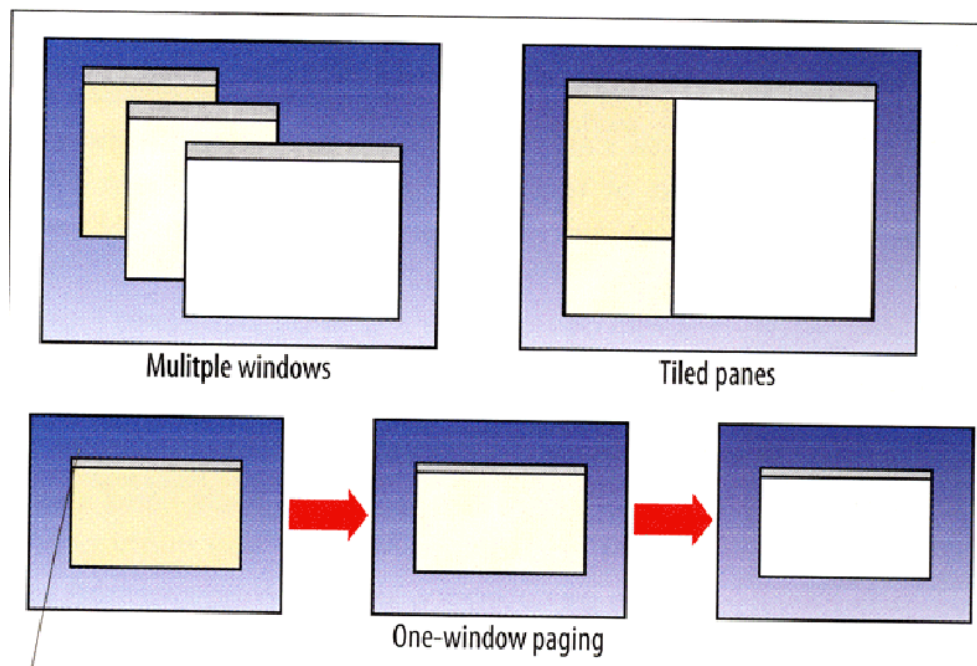
Some transform rules are explicit

Some transform rules are implicit or ambiguous or misleading

---

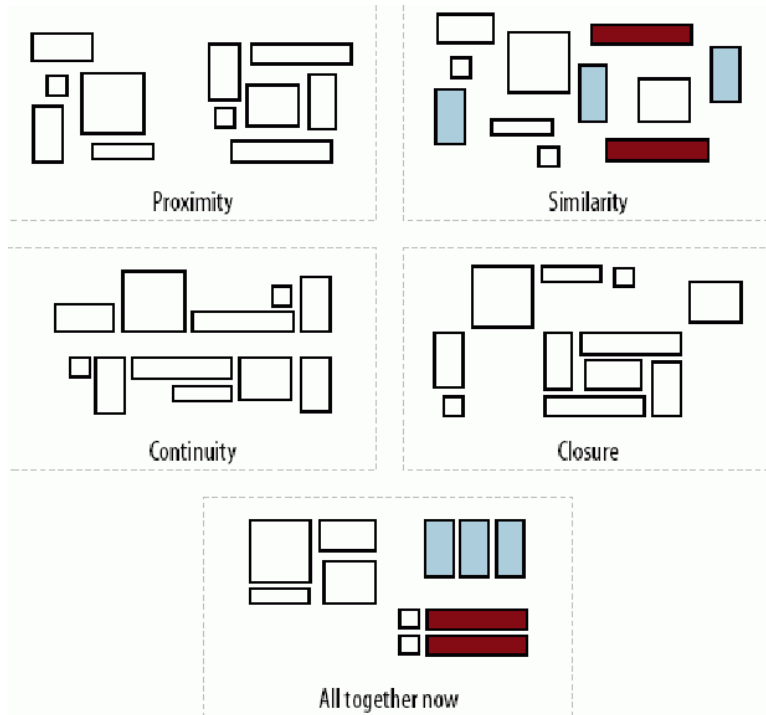
## Binding Structure to Presentation - Alternatives

---



# Gestalt Principles to Reinforce Structure

---



## Internationalization

---

Internationalization is the design of products and services so that they can be easily localized for specific languages and cultures

Many of the architectural and design principles for internationalization are specializations of the "mother" principle to separate content from presentation:

- Leave text outside of graphics
- Separate strings/labels from code in scripts
- Use variable names that are not words in the scripts

# Localization

---

Localization is the process of adapting the content and applications of a product or service to make it acceptable for a particular cultural or language market

Translation is the primary localization activity

The ease / automatability of localization depends on the principles and techniques used in internationalization

# User Interface Design Patterns

---

A very promising approach to making UI design more efficient, systematic and predictable

Patterns are solutions to common UI design problems that embody best practices for using structures and presentations for specific types of content

You can think of them as a very high level design vocabulary that guides designers toward better solutions than they would build if they started from less principled and tested starting points

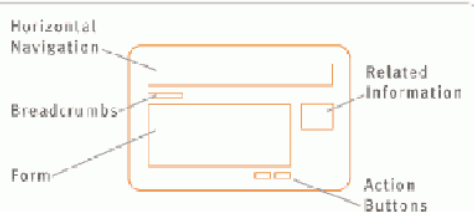
Pattern libraries help even experienced designers, but provide the most value for relatively inexperienced ones

# The UC Berkeley UIDP Project

---

**What's a Web Design Pattern?**

**A pattern is a description of a common web design problem and good solutions for that problem.**

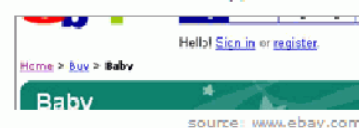


Horizontal Navigation  
Breadcrumbs  
Form  
Related Information  
Action Buttons


[Developer Resources](#) | [How to Use Patterns](#)

**Example Patterns**

**Breadcrumbs**  
The user needs to know how they have gotten to their current location within the Web site or application.




source: [www.ebay.com](http://www.ebay.com)

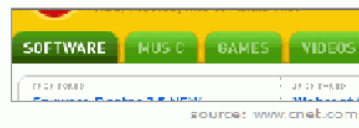


source: [www.walmart.com](http://www.walmart.com)

**Navigation Tabs**  
The user needs to locate information on a Web site that contains many categories and a complex hierarchy.



source: [www.cheaptickets.com](http://www.cheaptickets.com)



source: [www.cnet.com](http://www.cnet.com)

# The UC Berkeley UIDP Project

---

I-School '06: Kelly Snow, Mano Marks, Tim Dennis, David Hong

Target audience is UC Berkeley web developers, most of whom lack time or formal training in UI design

Very systematic and well-documented project to:

- Understand the skills and design practices of intended user community
- Analyze the kinds of applications they build
- Analyze existing pattern collections and "metamodels"
- Develop a methodology for developing patterns
- Develop 21 patterns in 5 categories (forms, information organization, navigation, profile management, and search)
- Design and test the user interface to the pattern repository
- Build and deploy the pattern repository

# Readings for Lecture #20

---

Elliott Rusty Harold, "Why XForms"

Kamlesh Pandey, "Business Rules in User Interfaces"

Dongsong Zhang, "Web Content Adaptation for Mobile Handheld Devices"

DON'T BOTHER TO READ: Paul Demery, "Beyond CPFR"