Usability Pattern-Oriented Design:

Maximizing Reusability of Pattern Languages over the Web

Homa Javahery

Human-Centered Software Engineering Group

Faculty of Engineering and Computer Science

Concordia University

Outline

- **General Overview**
  - ◆ **Patterns as a Design Solution**
  - ◆ **Usability Pattern Examples**
  - ◆ **Pattern Languages**
- **Research at Concordia University**
  - ◆ **MOUDIL**
  - ◆ **UPADE Environment**

Tools for Gathering and Disseminating Design Knowledge

- Aim to capture best practices about the design of usable systems

- Most used tools: guidelines and patterns

- Guideline Example: Dialog boxes from the Macintosh Human Interface Guidelines
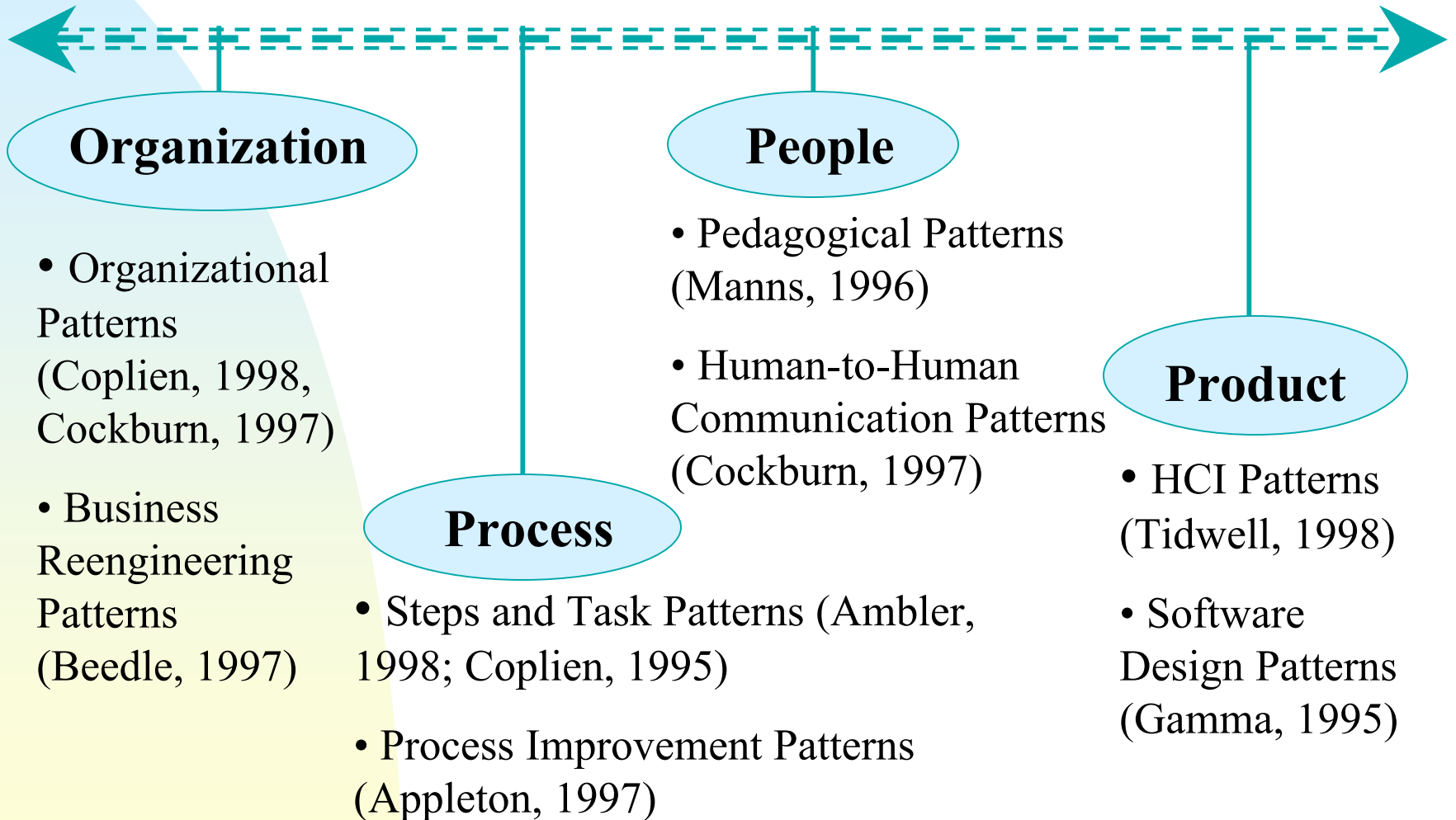 - 4 types
  - Eg. Alert boxes
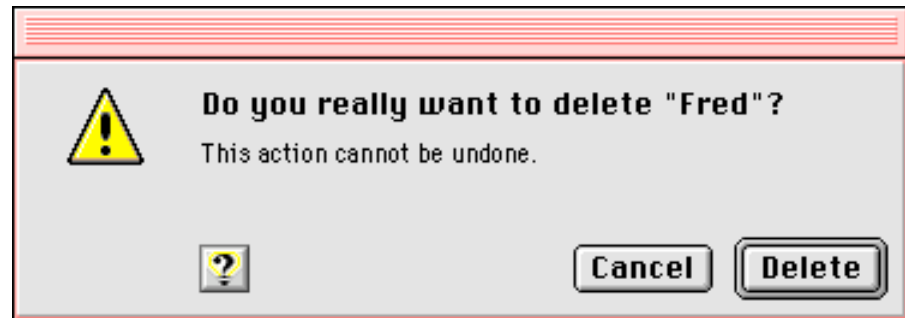
Disadvantages of Guidelines

- Numerous and difficult to select appropriate guideline for particular design problem

- Difficult to apply, especially for novice designers

- Guidelines contradict each other at times

Patterns as Alternative Design Tool

- A pattern is a *proven solution* to a *common problem* in a *specified context*

- Good alternative to guidelines -- concrete, easier to use

- Focus on context

- Tells the designer when, how & why

Software Engineering Patterns Mania

Organization

- Organizational Patterns (Coplien, 1998, Cockburn, 1997)

- Business Reengineering Patterns (Beedle, 1997)

Process

- Steps and Task Patterns (Ambler, 1998; Coplien, 1995)

- Process Improvement Patterns (Appleton, 1997)

People

- Pedagogical Patterns (Manns, 1996)

- Human-to-Human Communication Patterns (Cockburn, 1997)

Product

- HCI Patterns (Tidwell, 1998)

- Software Design Patterns (Gamma, 1995)

Example 1: Go Back to a Safe Place

Problem

- Users often get lost when exploring

- Users often want to go back to previous state they were happy with

- Users more likely to explore if they feel safe

- Backtracking out of a long navigation path can be tedious

Solution

- Provide a way to go back to a checkpoint of the user's choice

Example 1: Pattern Definition

Pattern Name: Go back to a safe place

Pattern Description:

- Specific example → *Home* button
- **Context → move through steps** ******
- Problem → how to make navigation easy
- Forces → users get lost, etc.
- Solution → go back to checkpoint

Usability Pattern – A Definition

- Distilled lessons learned and best design practices from real world experiences about how to develop usable systems

- Patterns expose knowledge that has been gained from different projects by many experts over many years

Example 2: Web Convenient Toolbar

- Provides a direct access to frequently used pages or services.

- Includes navigation controls for What's New, Search, Contact Us, Home Page, Site Map, and so on.

- Groups these actions together in a toolbar, labels them with terms or pictures whose meanings are self-explanatory, puts them in a fixed place so that they are easily accessible to a user visiting any page of the site.

Example 2: Web Convenient Toolbar

Advantages

- Reduce the cost of usability evaluation and testing

- Improve the reusability and standardization of applications

- Their assimilation, convey expert insight and knowledge to inexpert developers

- Help forge the foundation of a shared vision, and a collective of styles for developing usable products

Pattern Languages

- A collection of interrelated patterns
- Objective: to capture the collective wisdom of designers in a way that can be immediately used, especially by novice designers
- 2 criterion
    1. Standard Pattern Description
    2. Relationship between patterns
- If both criterion not fulfilled, it is not a language but a *catalogue* of patterns

Examples of Pattern Languages

- Experiences -- A Pattern Language for UI Design (Coram & Lee, 1996)

- The Brighton Usability Pattern Collection (University of Brighton, 2001)

- Common Ground (Tidwell, 1999)

- The Amsterdam Collection of Patterns in UI Design (van Welie, 2000)

Experiences

- Emphasis on the user's experience within software systems

- Patterns grouped by focus areas

- Pattern Description: narrative form using natural language

- Pattern Relationships: clear mapping of pattern interrelationships

Experiences -- Pattern Relationships
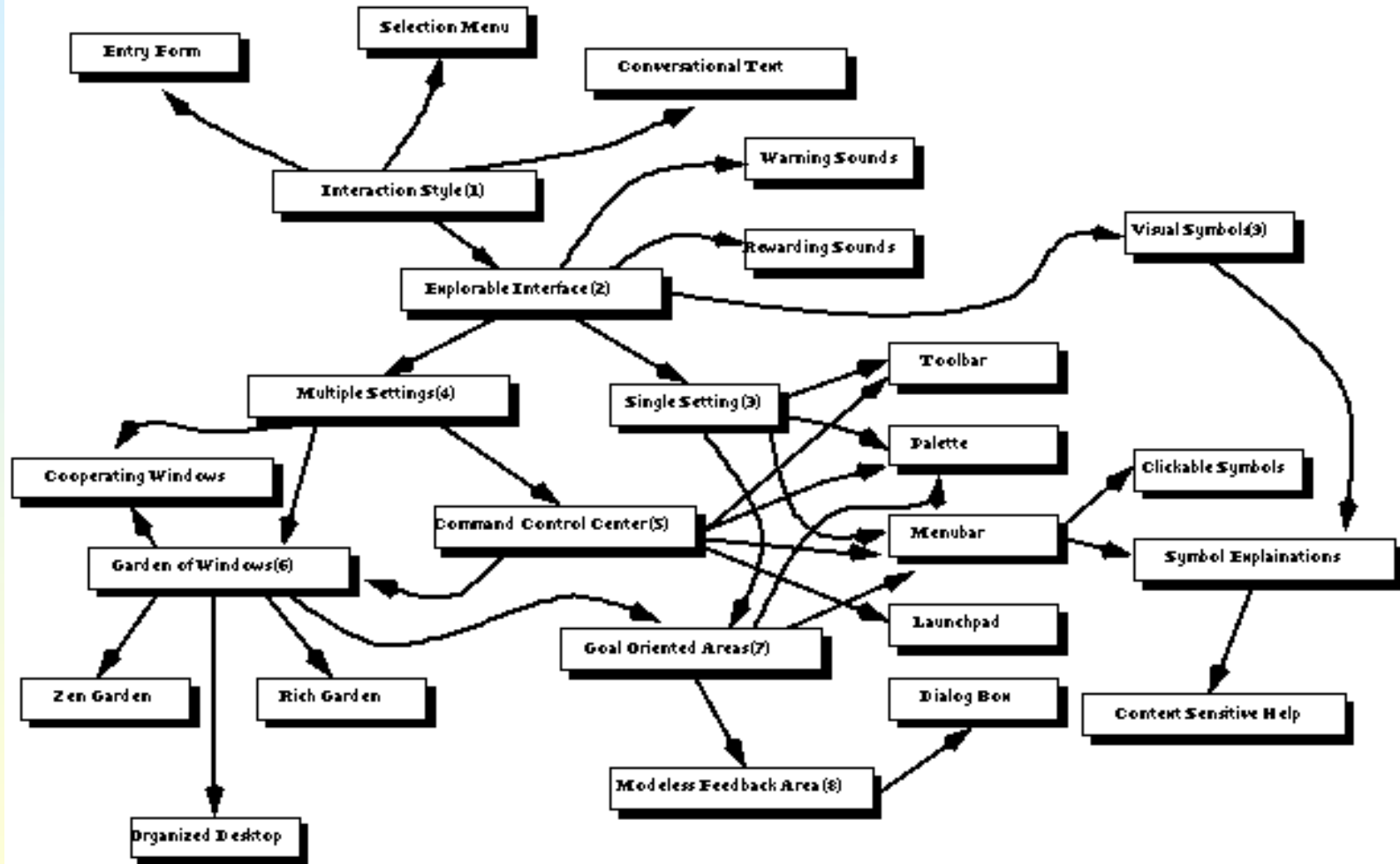
Problems with Pattern Languages

- Patterns...
  - There are NO standards for documenting patterns
  - We need tools to formally validate patterns
- Pattern Languages…
  - Need for formal language with syntax and semantics; not just narrative description
  - Need for context of use-oriented relationships

Projects in Development

- By the Human-Centered Software Engineering Group at Concordia University

- http://hci.cs.concordia.ca
  - MOUDIL -- Montreal Online Usability Digital Library
  - UPADE -- Usability Patterns-Assisted Design Environment
    - UPADE Web Language
    - UPADE editor

UPADE Environment
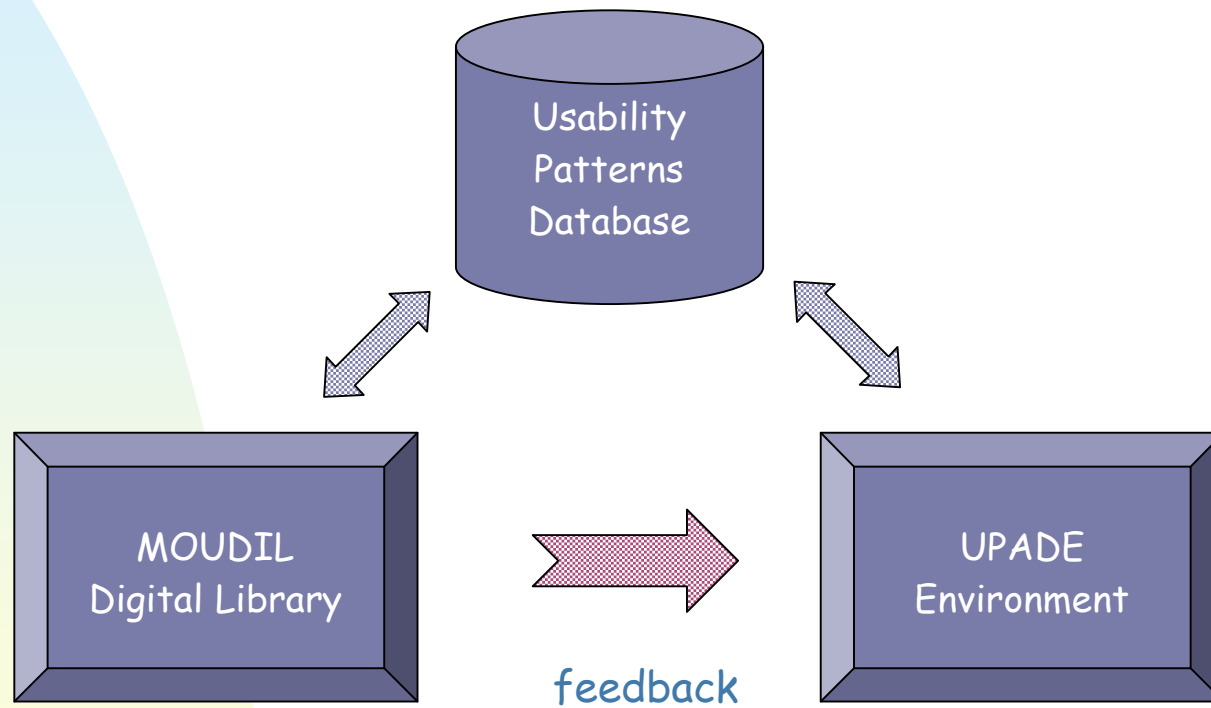
MOUDIL

- Montreal Online Usability Digital Library
- Online digital library for sharing usability and user interface pattern information

- Target: software and usability engineers, developers

- Objective: provide pattern information and receive feedback

- Functionality: Browse, Search, MySpace

MOUDIL Objectives

- Use feedback to answer certain questions about patterns and their uses

- How do developers create and document patterns?

- How are patterns used?

- Use this information to develop UPADE

UPADE Web Language Characteristics

- UPADE Web language is a usability pattern language for Web applications
- A set of usability patterns
- Patterns are interrelated
- Patterns are documented in formal notations
- Patterns are related to Web design process

Overview of UPADE Web Language
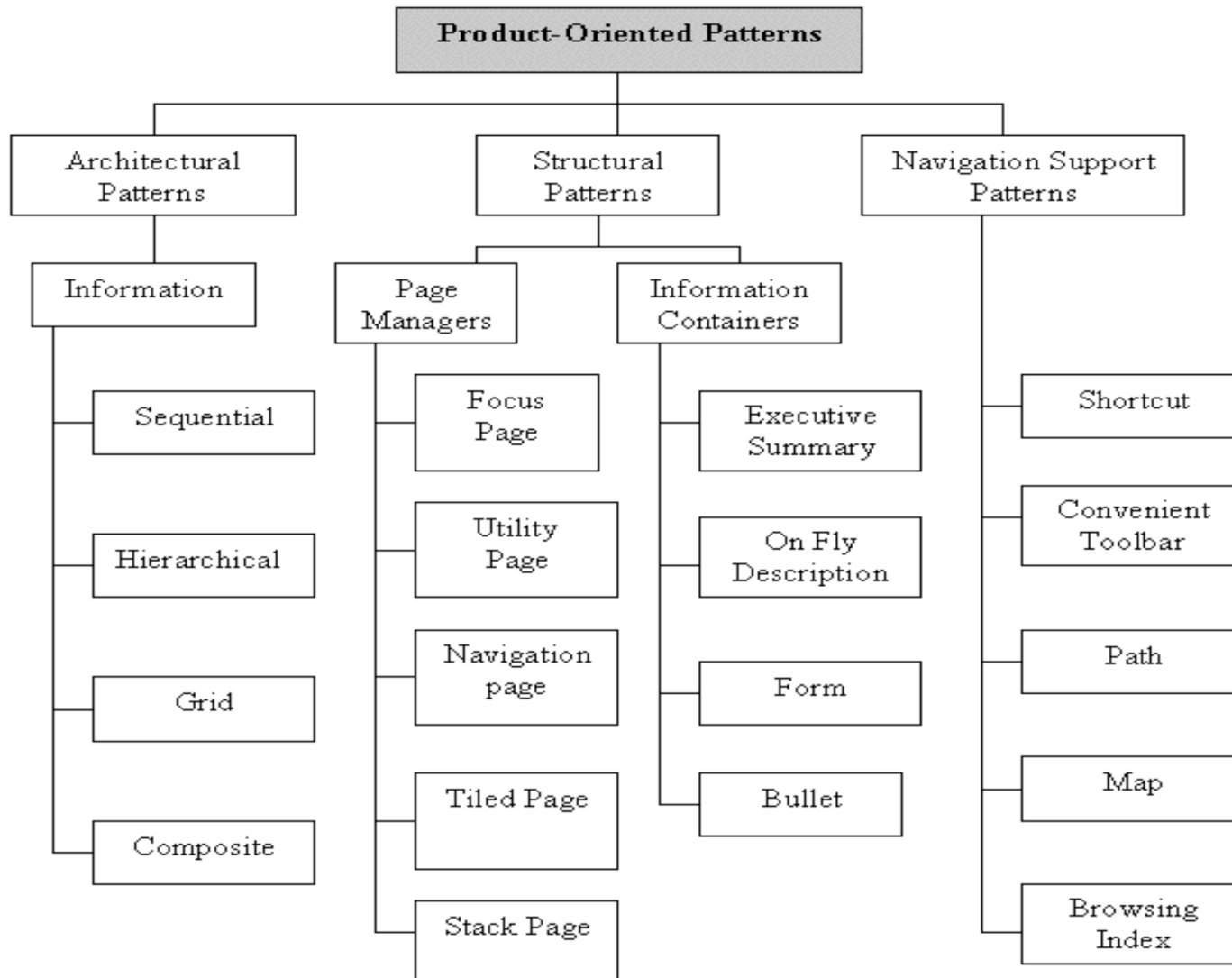
Pattern Categories

This language define 3 categories of product patterns:

- **Architectural Patterns:**
  - ◆ Describe different schemes for organizing the content of a Web application

- **Structural Patterns:**
  - ◆ Define physical and logical layout of commonly used web pages. Suggests how to group info in cognitively respectable structure and how to display them

- **Navigation Support Patterns:**
  - ◆ Provide navigation between a set of pages and segments of information

Hierarchical Pattern

- Architectural Pattern

- Pages are organized in a hierarchical model. The user
  can easily go from the most general overview of the web
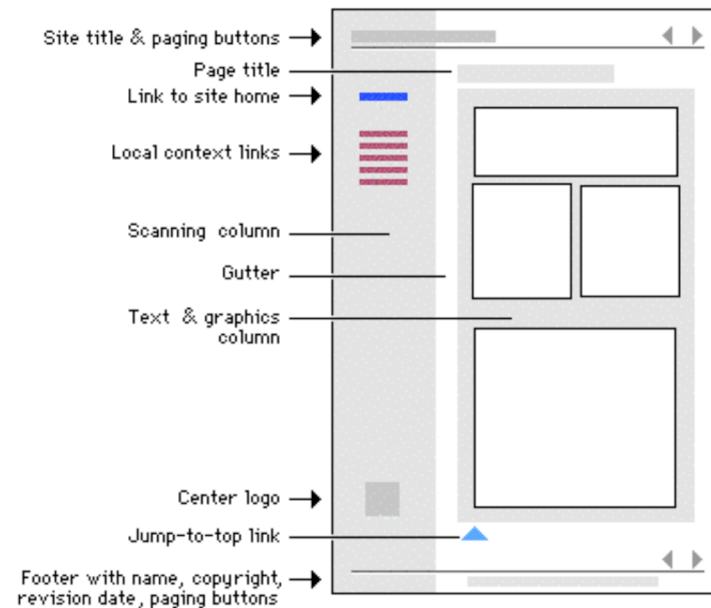 site, such as the home page, down to more specific topics

Composite Pattern

- Architectural Pattern
- A complex and large web application is generally organized using a combination of several architectural patterns
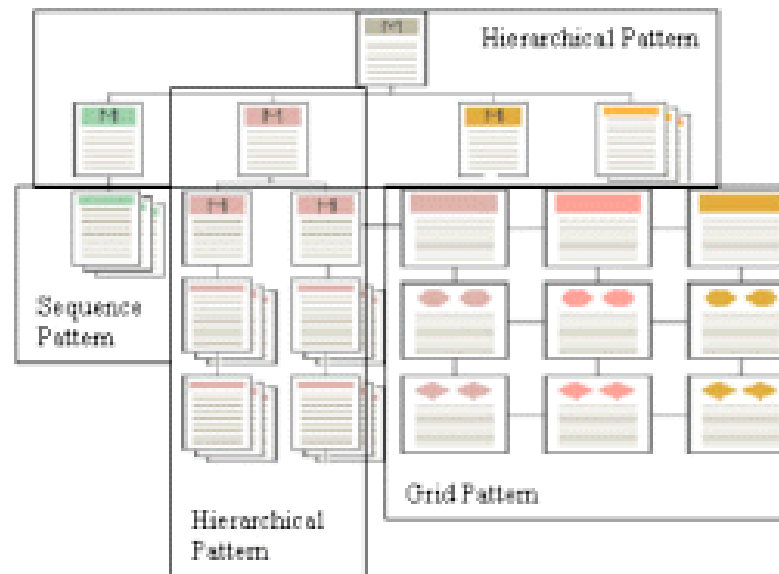
Focus Page Pattern

- Page Manager Pattern – subcategory of Structural Pattern
- Helps to build a Web page that is the fountainhead and center of the Web site. Balances aesthetics and practicality to attract users from their first glances

Site title & paging buttons →
Page title
Link to site home →
Local context links →
Scanning column
Gutter
Text & graphics column
Center logo →
Jump-to-top link
Footer with name, copyright, revision date, paging buttons →

Executive Summary Pattern

- Information Container Pattern – subcategory of Structural Pattern

- This pattern gives users a preview of underlying information before spending time in downloading and reading large amounts of information.

UPADE Web Language Format

- Pattern_Name: **Convenient Toolbar Pattern**
  - type of Navigation Support Pattern

- Context_Use
  - **User:** Expert
  - **Task**: Assist the user to reach the most useful and frequently visited pages at any time throughout the Web site.
  - **Workplace**: Web applications

- Usability_Problem:
  - The user can easily find the most commonly used pages regardless of the current state of the artifact.
  - The user can reach these convenient pages promptly.

Example - Convenient Toolbar Pattern [2]

- Usability_Factor
  - **Factor**: Efficiency, Safety
  - **Criteria**: Consistency, Minimal Action, Minimal Memory, User Guidance, Helpfulness

- Example

Example - Convenient Toolbar Pattern [3]

■ Design_Solution

  ◆ Group the most convenient action links, such as home, site map, help and etc.

  ◆ Use meaningful metaphors and accurate phrases as labels

  ◆ Place it consistently throughout the Web site

Example - Convenient Toolbar Pattern

Other Language Attributes:

- Design_Principle
- Related Usability_Patterns
- Reading

Final Remarks

- Patterns at different levels of the organization

- Usability Patterns as an alternative design solution

- Pattern Language
  - Pattern Definition
  - Pattern Interrelationships

- UPADE Web Language

- MOUDIL Digital Library – receive feedback

- Our website: http://hci.cs.concordia.ca

Questions...