Human Machine Interaction
B.E Sem VIII

Window Management

Learning Objective: To understand different Window Management schemes

SAKEC

Window Management:

Can be done using

- A single-document interface
- A multiple-document interface
- Workbooks
- Projects

1. Single-Document Interface

- A single primary window with a set of secondary windows.
- Proper usage:
  - Where object and window have a simple, one-to-one relationship.
  - Where the object’s primary presentation or use is as a single unit.
  - To support alternate views with a control that allows the view to be changed.
  - To support simultaneous views by splitting the window into panes.
- Advantages:
  - Most common usage.
  - Window manipulation is easier and less confusing.
  - Data centered approach
- Disadvantage:
  - Information is displayed or edited in separate windows.

2. Multiple-Document Interface

- A technique for managing a set of windows where documents are opened into windows.
- Contains:
  - A single primary window, called the parent.
  - A set of related document or child windows, each also essentially a primary window
- Each child window is constrained to appear only within the parent window.
- The child windows share the parent window's operational elements.
- The parent window's elements can be dynamically changed to reflect the requirements of the active child window.
- Proper usage:
  - To present multiple occurrences of an object.
  - To compare data within two or more windows.
  - To present multiple parts of an application.
  - Best suited for viewing homogeneous object types.
  - To clearly segregate the objects and their windows used in a task.

Conti..

- Advantages
  - The child windows share the parent window’s interface components (menus, toolbars, and status bars), making it a very space-efficient interface.
  - Useful for managing a set of objects.
  - Provides a grouping and focus for a set of activities within the larger environment of the desktop.

- Disadvantages
  - Reinforces an application as the primary focus.
  - Containment for secondary windows within child windows does not exist, obscuring window relationships and possibly creating confusion.
  - Because the parent window does not actually contain objects, context cannot always be maintained on closing and opening.
  - The relationship between files and their windows is abstract, making an MDI application more challenging for beginning users to learn.
  - Confining child windows to the parent window can be inconvenient or inappropriate for some tasks.
  - The nested nature of child windows may make it difficult for the user to distinguish a child window in a parent window from a primary window that is a peer with the parent window but is positioned on top.

3. Workbooks

- A window or task management technique that consists of a set of views organized like a tabbed notebook.
- It is based upon the metaphor of a book or notebook.
- Views of objects are presented as sections within the workbook's primary windows; child windows do not exist.
- Each section represents a view of data.
- Tabs can be included and used to navigate between sections.
- Otherwise, its characteristics and behaviour are similar to those of the multiple document interface with all child windows maximized.
- Proper usage:
  - To manage a set of views of an object.
  - To optimize quick navigation of multiple views.
  - For content where the order of the sections is significant.

Conti..

- Advantages:
  - Provides a grouping and focus for a set of activities within the larger environment of the desktop.
  - Conserves screen real estate.
  - Provides the greater simplicity of the single-document window interface.
  - Provides greater simplicity by eliminating child window management.
  - Preserves some management capabilities of the multiple-document interface.

- Disadvantages:
  - Cannot present simultaneous views.

4. Projects

- Description:
  - A technique that consists of a container: a project window holding a set of objects.
  - The objects being held within the project window can be opened in primary windows that are peers with the project window.
  - Visual containment of the peer windows within the project window is not necessary.
  - Each opened peer window must possess its own menu bar and other interface elements.
  - Each opened peer window can have its own entry on the task bar.
  - When a project window is closed, all the peer windows of objects also close.
  - When the project window is opened, the peer windows of the contained objects are restored to their former positions.
  - Peer windows of a project may be restored without the project window itself being restored.
- Proper usage:
  - To manage a set of objects that do not necessarily need to be contained.
  - When child windows are not to be constrained.

Conti..

- Advantages
  - Provides a grouping and focus for a set of activities within the larger environment of the desktop.
  - Preserves some management capabilities of the multiple document interface.
  - Provides the greatest flexibility in the placement and arrangement of windows

- Disadvantages
  - Increased complexity due to difficulty in differentiating peer primary windows of the project from windows of other applications.

Organizing Window Functions

- Organize windows to support user tasks.
- Use primary windows to:
  - Begin an interaction and provide a top-level context for dependent windows.
  - Perform a major interaction.
- Use secondary windows to:
  - Extend the interaction.
  - Obtain or display supplemental information related to the primary window.
- Use dialog boxes for:
  - Infrequently used or needed information.
  - “Nice-to-know” information.
- Minimize the number of windows needed to accomplish an objective.

References:

- The Essential Guide to User Interface Design Second Edition, Wiley.
- An Introduction to GUI Design Principles and Techniques ,Wilbert O. Galitz