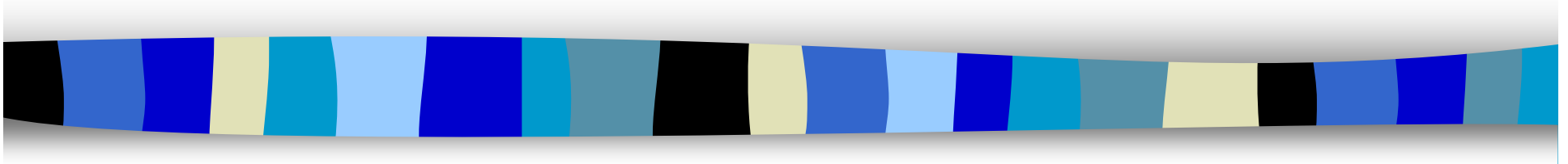


Interface in Games



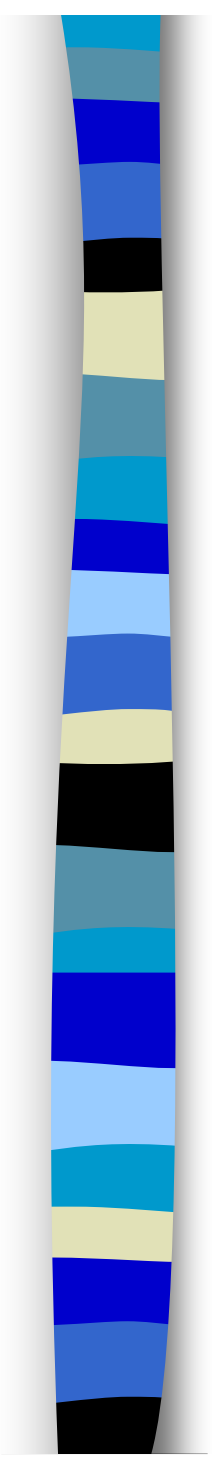
Topics in Game
Development

UNM

ECE 495/595; CS 491/591

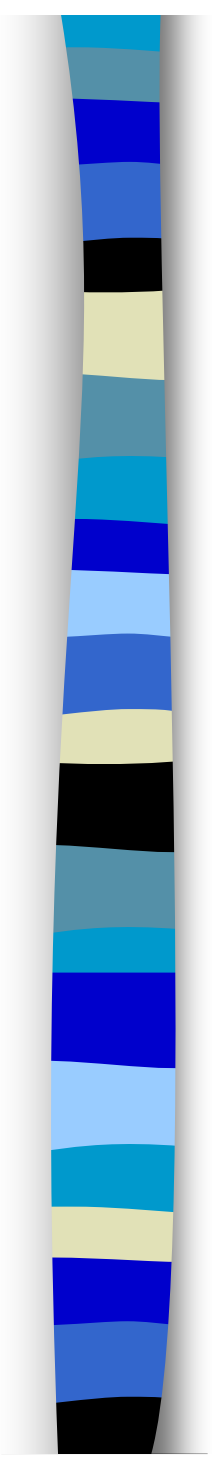
Feb. 27, 2007

© 2007 Carolyn Handler Miller



User Interface (UI) is:

- The connection between game & player
- How player receives information
- How player takes action
- One of the things that can make a game fun... or disappointing or frustrating



3 Major Types of Interface

1. Manual
2. Visual (active and passive)
3. Auditory

Manual Interface

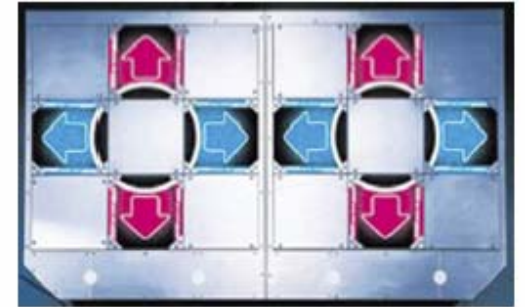
(Hardware Based Input Devices)



- Most common:
 - Controllers, joysticks for game consoles
 - Keyboard and mouse for computer games
 - Self-contained hand-held devices, mobile phones
- Each associated with certain game genres – pick platform/device best suited to your game

Other Examples of Manual Interface:

Bass fishing;
Samba de Amigo;
Dance Dance Revolution



One of the newest: the Wii



© 2007 Carolyn Handler Miller

And Yet Even Newer: The D-Box GP-100

- Designed for racing games, FPS, flight simulators
- Tactile feedback: acceleration, bumps, turns, stops
- Only \$14,999!



Visual UI: Active type

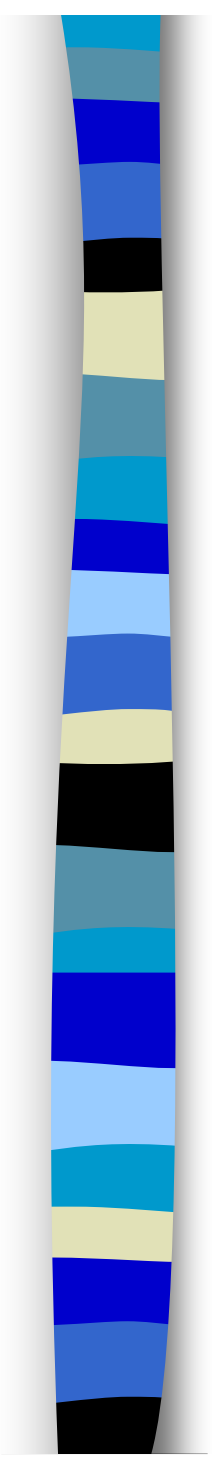
(Enables interaction – lets player do things)



Visual UI: Passive Type

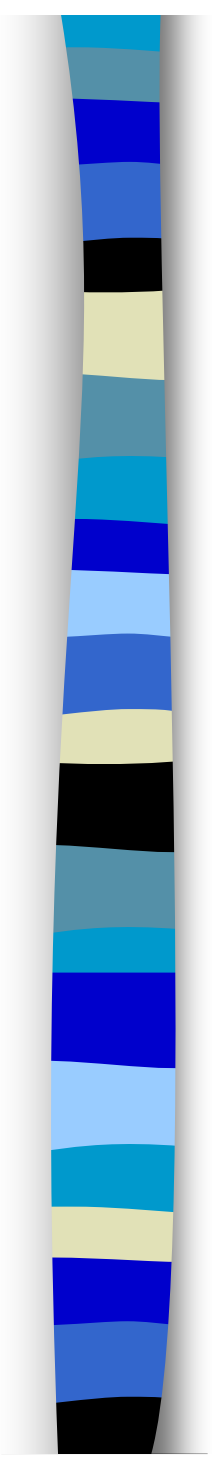
- Cannot interact
- Provides information like player status, location
- Information provided cannot be changed





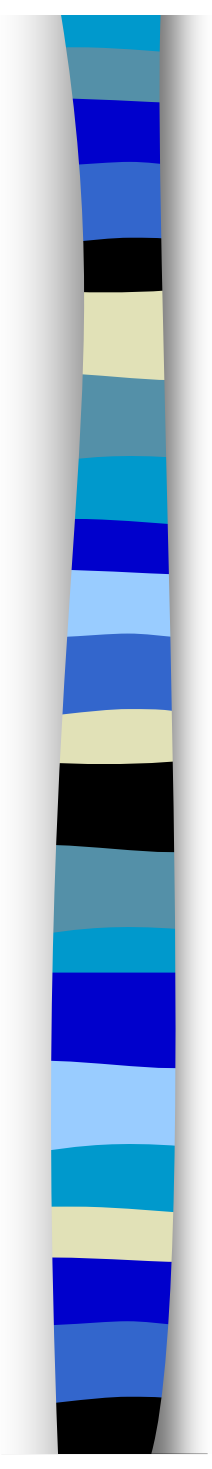
Visual UI Utilizes:

- Numbers
- Text
- Colors
- Icons
- Maps and other graphics



Auditory UI (Sound)

- May provide feedback when action accomplished: throw grenade, hear *bang*
- May cue player to do something: dog barks = time to hide
- May provide warning: rustling leaves = enemy near
- May be verbal feedback: instructions, praise, danger
- May offer clues about environment: tropics, city
- May be in form of music: cues, rewards



The Two Chief Functions of UI:

1. To give player information
2. To give player ability to act

UI Provides Information About All Important Aspects of Game



- Geography (maps, etc)
- Player status (score, health, skills)
- Inventory
- What player is doing
- If player is succeeding or failing

UI as a Way for Player to Perform Actions

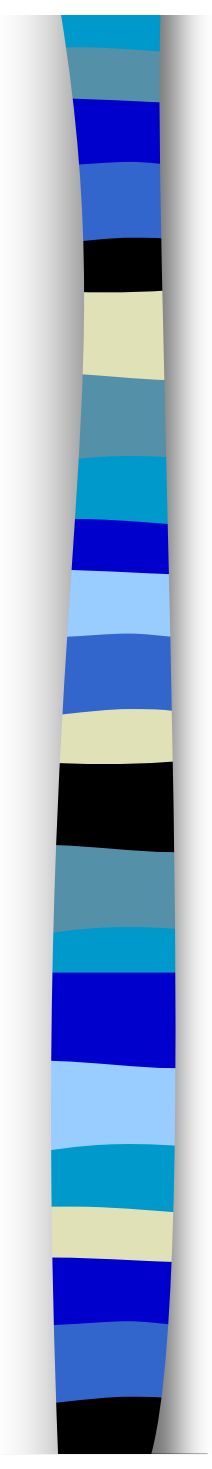


- Movements (run, jump, shoot)
- Navigation (travel long distances)
- Use of tools, weapons
- Customization of avatars, vehicles
- Interactions with NPCs, other players
- Collection of objects

Visual Interface Styles

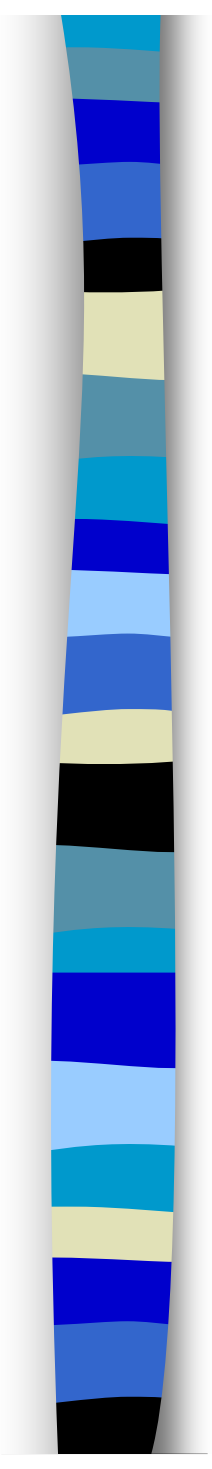


- Split screen (here); information can be on bottom, top, sides
- Windows
- Overlays (transparent, opaque)
- Virtually invisible



Visual Interface and Genre

- Visual interface varies greatly from genre to genre, depending in large part on types of actions players perform
- Best advice: study visual interface in other games in same genre as yours
- Model your game on others in genre
- Innovation NOT a plus in UI!

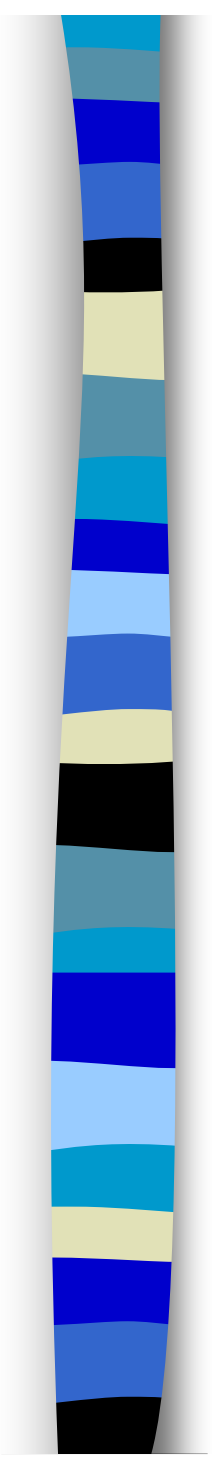


Saving as a Specialized Action

Saving raises issues of immersiveness, player control

Possibilities include:

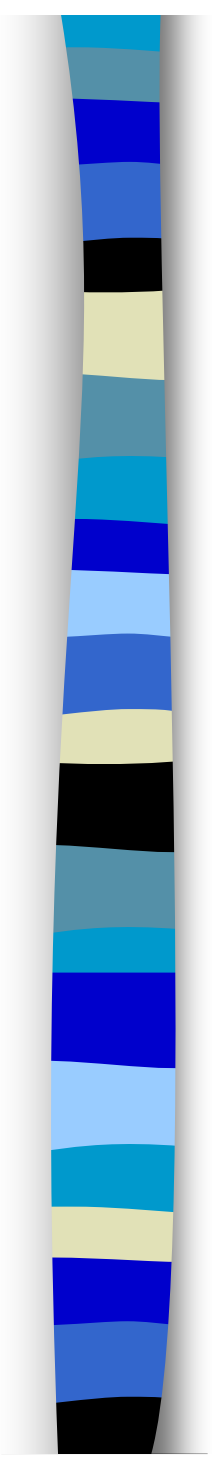
- Saving at automatic checkpoints (non-disruptive, but no player control)
- Save to file or save slot (offers player control but at expense of immersiveness)
- Quick save (non disruptive, but usually no options)



Taming Interface Complexity

(Thanks to Ernest Adams)

- Simplify, even if some authenticity is sacrificed (fuel-less car racing game)
- Automate some functions (car shifts automatically)
- Rein in numbers of simultaneous options, make player dig deeper (to change tires or adjust brakes go down several menu levels)



Ideal UI to Strive For:

- Easy to learn, easy to use; intuitive, clear
- Gives player helpful feedback
- Uncluttered, functional, efficient
- Doesn't strain player's memory
- Consistent
- Offers shortcuts to experienced players
- In aesthetic harmony with game world