

Structure in Games



Topics in Game Development

UNM

ECE 495/595; CS 491/591

Feb. 15, '07

What is game structure?



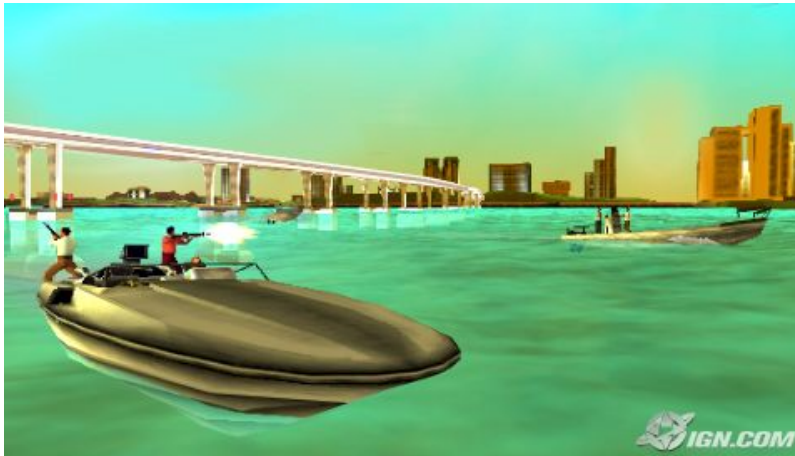
- Like other forms of structure: a framework
- The organizational underpinnings of the game
- Structure is NOT plot
 - Plot = the basic beats of a story, the “what happens next”
 - Structure = the underlying frame of the narrative or game

Why is structure important in games?



- Structure is like our bones -- without bones, would fall apart
- In games, structure is necessary...
 - To connect the various parts of game, to hold things together
 - To give game shape
 - For game to flow in satisfying way
 - For game to make sense

The Most Common Structural Form: The Level



- Levels are major subdivisions in a game
- Much like chapters in a book
- Often but not always accessed sequentially
- Genre specific terms:
 - Mission, quest, or chapter (adventure)
 - Match (sports)
 - Scenario (simulation)
 - Campaign (strategy)



Ten Considerations When Designing a Level

(with thanks to Ernest Adams, *Fundamentals of Game Design*)

1. What is its overall **function** in game? (to introduce a character, provide new challenge, reveal plot point?)
2. What is the **setting**? (what does it look like? What features does it contain?)
3. What is the level's **layout**? (where can the player go, and via what paths?)
4. What is the player's **main objective** here?
5. What are the **major challenges** and where will they take place?



Ten Considerations, Continued

6. What are the **initial conditions** of the level? (available resources, NPCs, gate opened or closed?)
7. What **narrative elements** does the level contain?
8. What are the **trigger points**? (What actions or decisions on part of player trigger what response)
9. What is the level's **mood or aesthetic style**? (and what, specifically, will help create it)
10. What are the **termination conditions**? (what constitutes winning or losing?)

Scoring High on Level Design

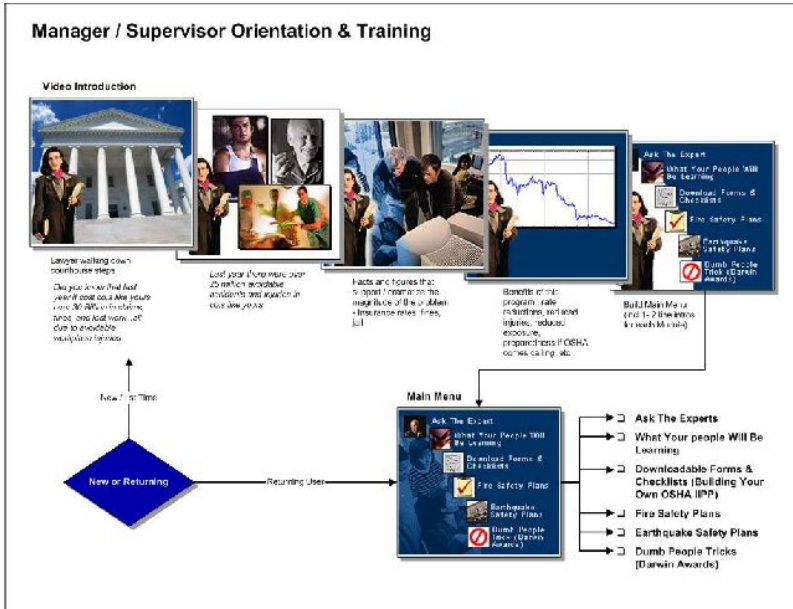
(also thanks to Ernest Adams)



1. Early levels as tutorials
2. Vary pacing within levels (action/rest)
3. Avoid non sequiturs – things that make no sense
4. Make level goals clear
5. Make possible consequences clear
6. Be generous with rewards, not with penalties
7. Always keep target audience in mind!

Other structural forms: The Module

- Usually found in games for education, training
- Usually select modules from main menu
- Usually can do in any order (after linear intro; often also a completion test)
- Modules may vary greatly in style within single game



Other Structural Forms:

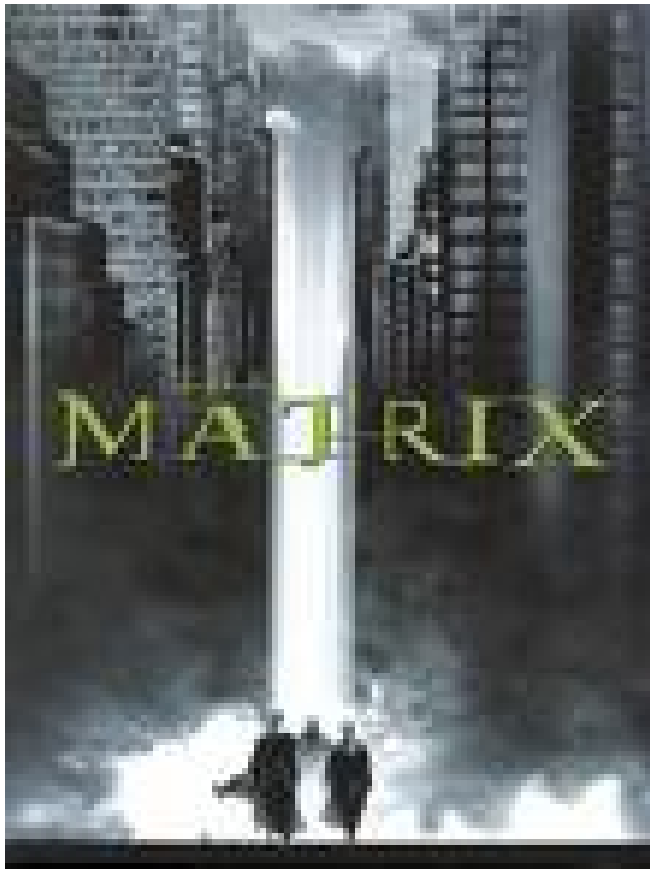
The Hub and Spoke



- Often found in kids' games
- Start from central location (hub); pick location (spoke)
- Once all tasks completed, may win a reward activity
- Very clear and simple to navigate, so highly suitable for kid's projects (though also used for adult games)

Other Structural Forms:

Parallel Worlds



- Same conceit as “The Matrix” movie
- Two or more worlds exist at same time... can jump between them
- Worlds are “persistent” – events continue in one while you are in another (MMOGs)
- Sometimes called “parallel streaming” or “harmonic paths”

Other Structural Forms:

The Sandbox



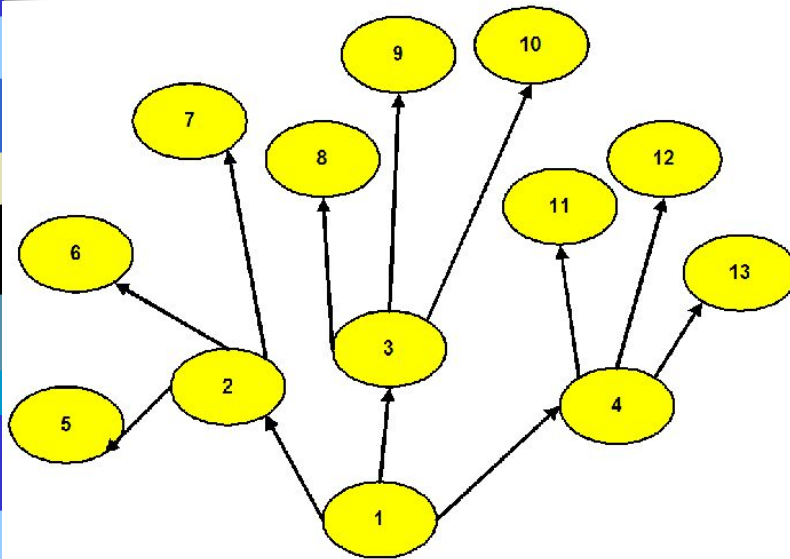
- As name indicates, very free-form
- Virtually no structure
- Provide player with objects, things to do, certain spatial boundaries
- No specific goals to achieve, no victory conditions

Creating Your Own Structure



- Sometimes necessary when no familiar models exist
- Example: *The Pig-Eating-Python* (linear at head and tail; highly interactive in middle)
- When inventing, consider objectives of game, target audience

What Underlies All Forms: The Branching Structure



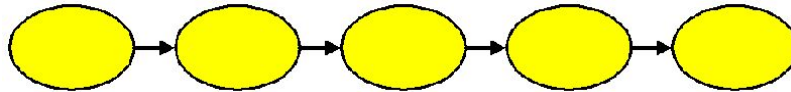
- Branching occurs whenever player faces challenge or must make a decision
- With every choice, 2 or more possible outcomes (“*if/then*” construct)
- Branching an inevitable attribute of all interactive works
- Problem: huge number of possible outcomes
 - Here: at 2nd tier: 13 possible outcomes
 - By 3rd tier: would be 39!
 - Most outcomes never experienced: branching gobbles up resources



Possible Ways to Rein In Runaway Branching

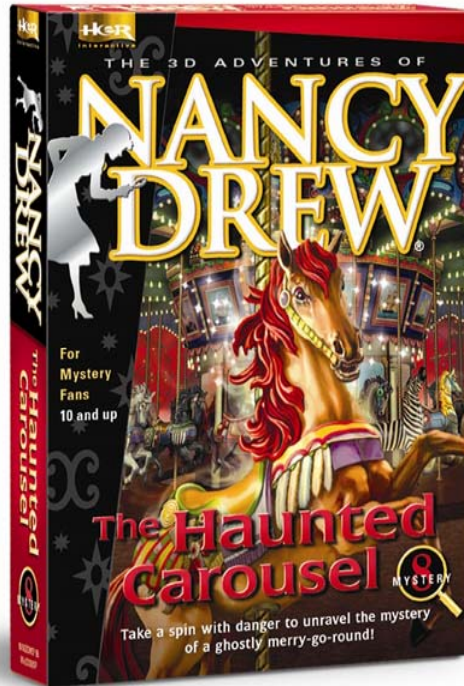
- **Cul de sacs**: areas open to explore, but dead ends; force player back to main path
- **Barriers**: block entry to an area until player performs “gateway task” – an activity that unlocks barrier
- **Loop backs**: forces you back to an earlier point

Another Technique: The String of Pearls



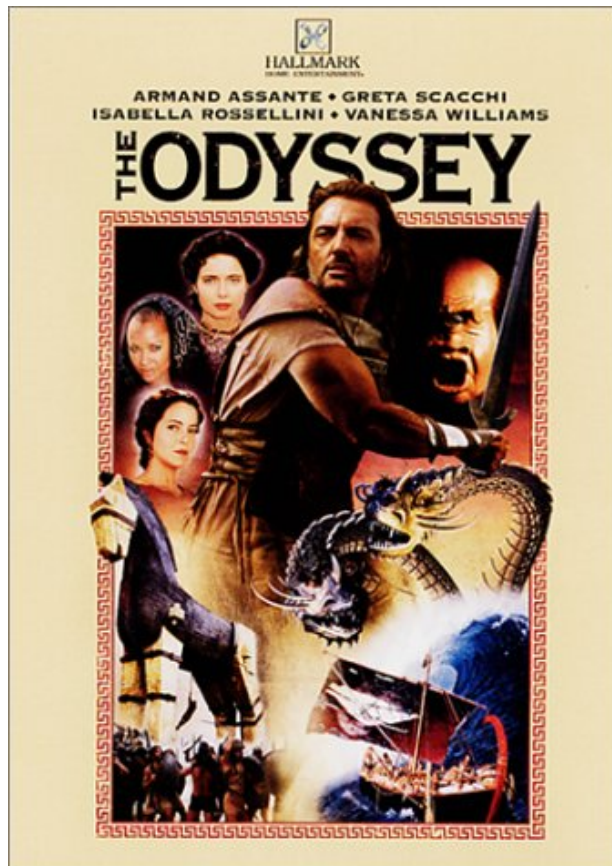
- Each pearl is a “world” that player can explore freely
- However, player often cannot move on to new pearl until all challenges in current pearl have been met
- Player passes through pearls one by one, usually in linear sequence (*sequential linearity*)

The Critical Story Path: For Story-Rich Games



- Useful in games where you want things to happen in some rough order (mysteries, etc.)
- A process by which you determine:
 1. All **critical story beats** (necessary narrative events)
 2. What needs to be conveyed via **cut scenes**
 3. What **essential actions** must be performed to serve as **triggers** for the critical story beats
 4. The **Non-essential** interactive opportunities you will offer

The Journey Story: As Tool to Convey Narrative



- Best known (but not only) model: the Hero's Journey, first articulated by Joseph Campbell
- Serves games well: has built in drama, challenges, settings, characters
- Plot advances as player moves to a new location, meets a challenge, makes a decision
- Often used in RPGs, adventure games

Exercise:

Designing a Level



- Rough out a game with Valentine's Day theme
- Pick one level and determine:
 1. Its function in game
 2. Its narrative elements
 3. Its setting and major features
 4. Player's goal in level
 5. Main challenges
 6. Termination conditions