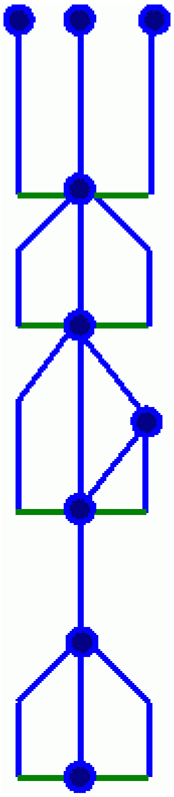
Scriptwriting for Games: Part 2: Advanced Plot Story Structures

Parallel Paths

The parallel path approach (which could also be considered a fully recombined branching structure) is a tenable and useful approach to story design, used to some extent in games such as Konami's *Metal Gear Solid* (1998), and Ion Storm's *Deus Ex* (2000). Branches recombine at the key story points, creating a structure which is a balance between linear and branching structure:

In the following diagram, those story points which have a green line through them are branch points splitting into three routes - but those paths then recombine before the next main story point. The underlying structure is linear:



every player must through every point on the spine of the story, but there is still some freedom as to how the player reaches the next point.

Metal Gear Solid, whilst essentially a linear game, features a single segment of parallel path structure which does much to improve the player's involvement in the game's story. This sequence, in which the player is tortured repeatedly but has an opportunity to escape between the torture sessions, has three paths through it: the player escape in a couple of different ways from the cell between torture scenes, they may admit defeat to the torturer, or if all else fails, they are eventually rescued from the cell by a deus ex machina. This parallel path segment also governs a split in the games ending: if the player does not give up in the torture scenes, they will end up rescuing the romantic interest - but if they do, she is killed and the player ends up escaping with a buddy character instead.

Although simplistic, this remains a strong example of the power of even a basic non-linear structure, giving the player a genuine moral decision to face with consequences within the framework of the story.

This example also typifies the needs of the parallel path structure's branch points: a primary path (rescue the romantic interest), a secondary path (escape with the buddy character) and a catch-all tertiary path (the player is rescued) to ensure that the plot will move forward whatever happens. In point of fact, this sequence from *Metal Gear Solid* is two separate branches (resist torture, or give up is the first branch; escape on their own, or be rescued is the second branch) leading in effect to four different paths through this sequence, ignoring the issue of the means by which the player may escape by their own means. Only the first branch is expressed in the development of the rest of the plot, however, and in the branch at the end of the story.

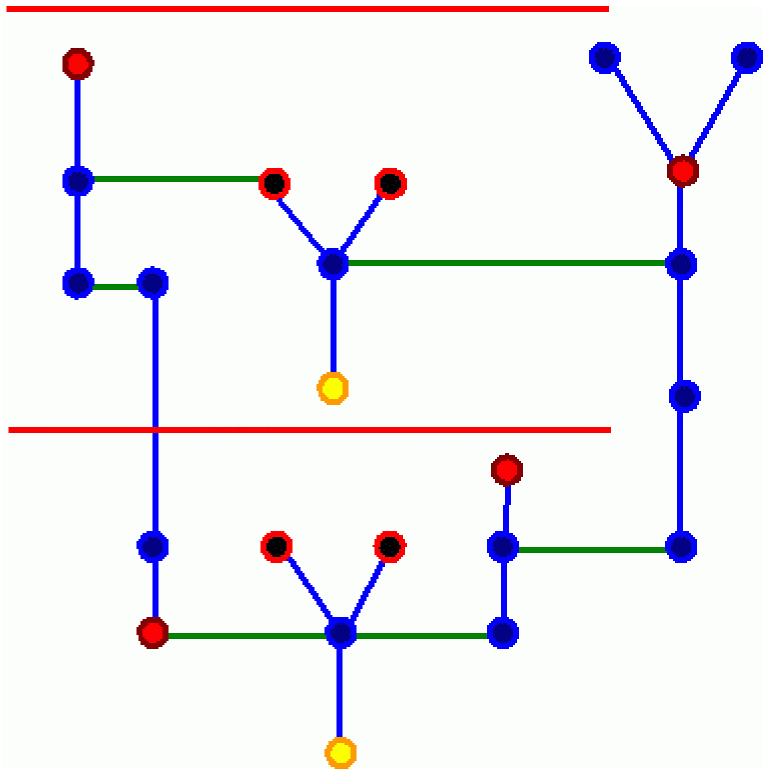
The disadvantage of the parallel path structure is that more is developed for the game than is actually seen. As soon as you have two endings, only players who play the game twice will have any hope of seeing all of the story material, although this disadvantage is offset if it is expected that most players will replay the game (i.e. the game is both short and fun).

The advantage lies in the choices the player has within the plot, and the sense of connectivity with the story that is engendered. Every time a game gives the player a meaningful choice, the become more emotionally involved with the story - although of course this may only be effective if the player knows they had a genuine choice. Because many players do not expect to be given a meaningful choice, sometimes the player will play along with what they believe is expected of them, because their prior experience of games has given them no reason to believe that they are facing a genuine decision. It is important, therefore, to ensure that the player is subtly taught or informed that their actions will affect the outcome, if this is in fact the case.

Threaded

The threaded structure is a powerful, complex way of allowing the player a non-linear story experience. Games such as Perfect Entertainment's *Discworld Noir* (1999) use a threaded structure to give the player a genuine degree of freedom to explore the story space once the initial formalities of establishing setting, character and interface have been concluded.

The following diagram is an approximation of the structure of *Discworld Noir's* first two Acts:



Unlike previous diagrams, the course of the plot does *not* follow a single path in this form of structure. Rather, the story is comprised of a number of different threads that develop largely independently (at least until the closing stages of the story). This parallels the methods that quality novels and films employ to develop the plot, with multiple narrative threads being woven together to produce a satisfying story.

The key to the symbols employed is as follows:

- Yellow circles represent the events at the start of an Act. Every Act starts in the same way in this particular example of the threaded structure, although this is not a given
- Blue circles represent non-mandatory plot events
- Red circles represent mandatory plot events: the player must experience all red circles (and must experience at least one circle with a black centre) to proceed to the next Act.
- Red lines show Act boundaries.
- Green lines show the places at which threads connect to one another (the places at which new threads are 'unlocked').

For example, starting at the beginning of Act I (the bottom yellow circle), the player is introduced to some back story and placed in the role of a private investigator in a fantasy world. They begin their investigations and soon reaches a point at which they are investigating multiple cases (the green line opens up two other threads, one to the left and one to the right). The player can eventually conclude the first stages of their initial investigation in two different ways (the red circles with black centres), but they may also begin investigating two other (apparently) separate cases (the path to the left and the path to the right). The right-hand thread represents the true conspiracy the player is investigating, and by the end of the first act, the player has already changed their conception of what is going on twice (signified by successive jumps along green lines).

The left hand path (which represents a side-story concerning a missing dwarf) shows an optional side path in which the player may investigate the case to a certain extent in the first Act, but cannot conclude it until the second Act. This case can also be largely side-stepped, as another chain of events in the second Act will eventually connect with the details of this case allowing it to be ignored - at least until it closes in on its terminal red circle which is an essential part of the plot.

Although this system is highly expressive, and leads to a freedom of movement within the story space which is uncommon, there are several problems which makes this approach of limited future value. Firstly, tracking changes in the plot skeleton during development is nightmarishly difficult: because of the sense of interconnectivity between events, it is very hard to implement structural changes. Furthermore, the complexity of the plot skeleton means that very few members of the development team will fully understand the structure. Given how difficult it is to fully appreciate a diagram showing a simplified version of the structure, it is easy to see why dealing with this structure in practice is no small undertaking.

Additionally, the degree of variance in the story structure makes QA much harder to perform completely (especially if there is no way to educate the QA team on the full extent of the underlying plot structure) which can lead to an extended QA period being required to ensure the stability of the code.

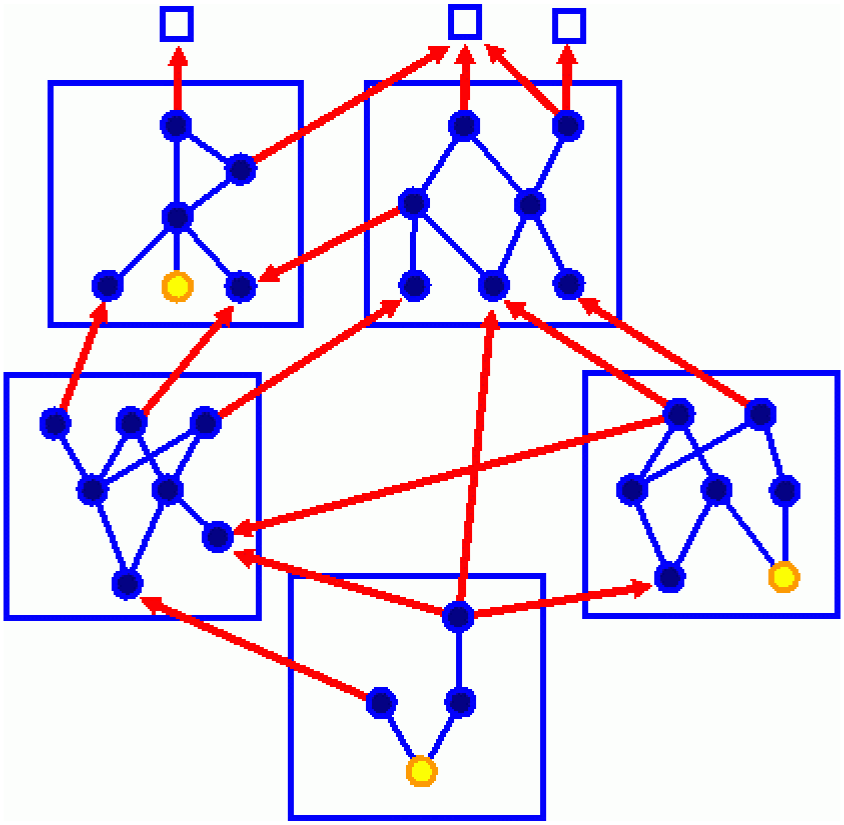
Of course, these problems are offset by the highly expressive nature of the structure, and the number of options for player freedom within the story. Unfortunately, much as with the parallel path structure, a great deal of this is lost on players if they do not realise that the freedom exists. Unlike the parallel path structure, however, the threaded structure has an ace up its sleeve: with so many routes through the plot supported, the player may not recognise the non-linear nature of the story, but they find that the game supports a fair degree of the actions they want to perform within the plot, which aids immersion. In essence, the advantage comes in the player not being blocked in their attempts to experiment - they are never faced with the inevitability of conforming to a linear plot (at least until the threads begin to recombine prior to the ending).

Nonetheless, what we really desire for interactive plots is a system as expressive as a threaded structure but as robust as parallel path structure.

Dynamic Object Oriented Narrative

A possible future of interactive plots is suggested by systems for dynamic object oriented narrative such as International Hobo's *FreeSpeak*. These use a notion from computer science (that of object orientation) with the framework used in commercial television series to provide structures that are even more expressive than the threaded structure, yet as robust as a parallel path structure.

The following diagram shows a hypothetical object oriented plot skeleton:



Before discussing the illustration, the idea of 'object orientation' must be approached. Simply put, object orientation concerns dealing with a problem by composing the data space from objects, which contain other objects, which contain other objects (like quarks in hadrons in atoms in molecules in bigger things). Everything is composed of small functional components, except for the atomic units at the base of the chain. The idea revolutionised programming languages in the 1980's, and may be about to do the same for interactive storytelling.

In the above diagram, blue squares represent an object known as an Episode (small blue squares indicate Episodes not covered in this example diagram), while all circles represent a Scene. Each episode consists of a number of Scenes which can use any structure - linear, branching, parallel path etc. Because each Episode is comprised of a small number of Scenes, the resulting set-up is robust and connectivity can be tracked on two different levels: transitions between the Scenes of an Episode (blue lines), and transitions between Episodes (red arrows).

Episodes are broadly chronological, but Episodes that refer to events that occur concurrently (i.e. that are at the same height in the case of this diagram) can occur in any sequence. In this particular example, the possibility of starting the story at different 'places' in the plot is represented by yellow circles (which might, for instance, represent playing through the story with a different character who joins the events at a different point).

Obviously if each Scene were a unique 'game level', this would suffer from many of the problems associated with multiple paths: developing many more resources than will typically be seen by the player. This is solved in *FreeSpeak* by the use of various other component entities, namely Personas (characters) and Locations. Every Scene takes place between certain Personas in certain Locations, and may be interactive or a cut scene. A particular Location may be used for many different game play and story related activities. For example, in Sega's *Shenmue* (2000) the docks are used as a stage for investigation, cut scenes, fighting, quick time events (interactive cut scenes) and even racing (although it should be noted that this reuse of location was performed in a linear fashion in *Shenmue*).

Any key gameplay that the designers wish every player to experience can be built into the plot skeleton as an event that must occur at some point - but the object oriented structure allows for that point to occur at many different places within the structure of the plot. The rest of the time the Locations are being used as a setting for basic repetitive game play (fighting, exploring, investigation etc.) using the core game engine. The design of a location is thus built around its uses in the game, but it can also be reused for story purposes.

This parallels a TV show which has a number of sets which are reused week by week as the core places the story takes place. A Location in *FreeSpeak* is analogous to a set in a TV show, with the same advantages in terms of cost-effective reuse of resources.

Once the initial investment in the creation of new tools, and the corresponding skills required to operate those tools effectively, has been overcome, the advantages manifest. The approach promises an extremely expressive and robust approach to interactive plots with a degree of freedom and level of replay value (exploring different renditions of the story) which would exceed anything we have currently seen in games.

Since its inception, developments in interactive storytelling have been gradual to say the least, although it is quite possible that a dynamic object oriented narrative system such as *FreeSpeak* will revolutionise the approach and create a new breed of games built around an advanced interactive plot.