The Importance of a Background: A Brief Breakdown

Game backgrounds come in many different varieties, each with their own pros and cons. A brief examination of some different types of backgrounds, as well as a sampling of the technologies they use during their implementation.

Games are more than simply sprites and sounds flying across your screen. Without a theme or goal, a game can seem empty and devoid or worth – the time spent on a game has to have some kind of driving force behind it. One of the many ways to do this is to create a theme or atmosphere for your game. Even a small amount of effort gives the player more investment into the game they're playing. One way to create this atmosphere is through the ambient visuals of the game, such as the background. The background of a game can be used as a key tool in establishing a game, both through gameplay and impact. This paper will provide a high level overview of three different types of background, their pros and cons, the types of technologies they employ, and some examples of games utilizing them.

Static Backgrounds

A static background is one which doesn't change during gameplay. In essence, it provides a simple backdrop for the player to play against. While this sounds simple and uninteresting, there is no limitation on how many static backgrounds can be used in the design of a game, making a level based system.

In order to create this effect, the background is drawn either as a collection of graphic primitives, sprites, tiles, or a larger image pasted onto the background. The background can represent barriers, doorways, or collisions, all depending on the code implementation – but it does not chage its state or move. Generally the background are designed in an image editing software, such as Paint, Photohop, or Picta.

Games which utilize static background include Solitaire, Space Invaders, and Pacman. A large-scale example of multiple static backgrounds would be the old Zelda series games, where each room is a unique background, and a new room is loaded when the edge of the screen is reached.

Scrolling Backgrounds

Scrolling backgrounds have more variety than the name suggests. The common thread to all scrolling backgrounds is that they adjust as the player moves through the level, and represent a much larger world than a static background, and that they all require some kind of rendering engine be supported to draw the images to the screen. This can be acheieved in a number of different ways, and each way provides different pros and cons.

Tile Based (pure): these backgrounds are implemented using a strict tile based system, and enforce gameplay mechanics to operate along these restrictions (meaning strictly upwards, downwards, left, and right). These maps are often simple to create and code, but are restrictive for the player and require a level of ingenuity to pull of in a creative way. The map can be saved as a large image of assembled tiles, or created using some software such as Mappy. An example of a game utilizing this level system is the original Prince of Persia game.

Tile Based (smooth): these backgrounds are also made through the assembly of many tiles, but allow mechanics which arn't strictly bound to the compass directions. This could include ramps, angles shots, wall jumping, and more, depending on the implementation of game. This is one of the best options for 2D sidescrolling games, as it allows a diverse range of mechanics and player interactivity while remaining relatively simple to program and develop. A map editor works best for implementing this type of game, allowing custom effects on each tile. Games utilizing this effect include MegaMan and

Street Fighter.

Bitmask: Bitmask style backgrounds are similar to smooth tile-based games, but rather than using tiles to create the map, the utilize an image to determine the collisions for each pixel. This allows for a smooth, pixel-to-pixel adjustments of the image at the cost of increased complexity and memory usage. This is done by considering each pixel a 'tile' and running the same algorithm as the smooth tile based system. Games such as Worms and Vectorial utilize this approach.

Vectorial: A vectorial map uses a combination of lines and polygons to create the boundaries for collision areas. This is similar to the Bitmask style maps, but requires much less overhead (no pixel-to-pixel computation) at the cost of a less intuitive way to construct the levels. Due to the construction of the levels being so unique, many of the algorithms and techniques used to detect slops, falling, etc. in the other approach fails to suffice and must be implemented uniquely for a vectorial map. Ori and the Blind Forest is a game using Vectorial maps to create its levels.

3D Backgrounds

3D backgrounds are all created using 3D rendering software as the worlds are built. The backgrounds are different from the 2D environments, as they can be viewable from many angles. Often, a 3D background is the same as the world the player operates in, but is unreachable due to some obstacle. Many modern games employ 3D worlds and backgrounds – almost all new generation fall under this catagory.

The breadth and scope of backgrounds plays a huge part in the diversity of our games. Without the evolution of backgrounds, all games would be tied to resembling Space Invaders or Pack-Man – great games in their own right, but limited in their design and scale. By expanding the backgrounds, you can create worlds, effects, and mechanics to help make a game stand out against its competitiors, or just be interesting to play in and of itself. The effort spent into creating a background will be well rewarded as it provides direction and thematic ties to a game.

REFERENCES

-3D Backgrounds:
https://books.google.ca/books?id=UZglGi-MDZIC&pg=PT81&lpg=PT81&dq=essentials+to+3d+backgrounds&source=bl&ots=u9SSM3iMDY&sig=Nkc6GBwg4CAFIsYVCf9_RAMw6Fo&hl=en&sa=X&ved=0ahUKEwiW_v-2uqjSAhUJ7oMKHav6BTsQ6AEIQjAH#v=onepage&q=essentials%20to%203d%20backgrounds&f=false

-2D Backgrounds:
http://higherorderfun.com/blog/2012/05/20/the-guide-to-implementing-2d-platformers/