Essay

Text

Richard A. Bartle

MAKING PLACES

The online computer games known as virtual worlds or MMORPGs (Massively Multiplayer Online Role-Playing Games) attract millions of players across the globe, each investing many hours a week in his hobby for months if not years. Few other forms of entertainment are this compelling, and yet MMORPGs remain little understood. There are theories as to *why* people play them with such dedication, but little discussion of what they *are*. In the past, they've been described as games, simulations, services and media. But fundamentally, they are none of these; virtual worlds are *places*.

As places, virtual worlds have a number of place-like features: they exist at all times, you can visit them and you can do things while you're visiting them. The main difference between virtual worlds and other places is that the former are not real – they're maintained entirely by computers and exist only in the human imagination. Formally, a virtual world is an automated, shared, persistent environment with and through which people can interact in real time by means of a virtual self; informally, it's an imaginary place able (through the magic of computers) to masquerade as real, such that you and other people can go there whenever you want and do things both in it and to it.

In real life, place is a natural consequence of space, which in turn is a given. In virtual worlds, this is not the case; rather space has to be *represented*. To date, there have been three main representations employed corresponding to three prevailing display formats: nodes for textual worlds, grids for isometric graphical worlds and polygons for 3D graphical worlds.

Text and Nodes

The first virtual worlds were text-based. Everything in them was described in words: the world, its inhabitants, the objects, the players, the events that occurred, the actions that the players undertook – everything! Almost all today's virtual worlds are directly descended from a single, textual primogenitor, *MUD* ("Multi User Dungeon"), which was written in 1978 (Bartle & Trubshaw 1978). *MUD* begat many imitators, one of which, *AberMUD* (Cox 1987), was released to the nascent Internet and soon afterwards became responsible for three new branches of virtual worlds: *TinyMUD* (Aspnes 1989), which eschewed game-like aspects and concentrated instead on the social side of things as well as world-building, *DikuMUD* (Nyboe et al. 1990), which emphasized strong, combat-oriented gameplay, and *LPMUD* (Pensjö 1989), which was very customizable and fell somewhere in between the other two. *TinyMUD* led to *LambdaMOO* (Cu 1990), the textual precursor to today's *Second Life* (2003), *DikuMUD* led to the majority of today's game-style worlds, such as *World of Warcraft* (2004).

Textual worlds represent space as a set of interlinked nodes. Each node embodies an atomic location (commonly called a *room*), which generally conceptualizes the smallest meaningful space into which a player's character can fit. This does *not* mean that a virtual world's rooms are all the same geographical size, nor that they are indoors: a mountain and a cupboard can both be valid as single rooms.

A room's description will usually place that room within a context, noting other rooms that are to be regarded as adjacent. These adjacent rooms can then be reached from the original room using primitive, directional movement commands. Typing a direction will move the player's character from the current room to the one pointed at by the appropriate exit link. A map for a textual world therefore consists of a network of rooms connected by a set of arrows that correspond to movement commands. Such an approach implies a fairly coarse granularity, but it is not one that unduly offends players' sense of locale.

This modeling of the virtual world as a network of nodes has some interesting properties, all of which are at times useful to designers:

The arrows on the map need not be bi-directional – north from room A may lead to room B, but south from room B could lead to room C.

Arrows need not connect different rooms. North from room A could lead back to room A.

Because rooms are just another kind of object, they can be picked up, carried around and placed inside one another – or, indeed, inside themselves.

Arrows can be changed dynamically to point at different rooms.

To implement a set of rooms networked in this fashion involves many data records connected by many more pointers. It is far easier to arrange rooms as a simple, two-dimensional array of squares; a grid 100 by 100 rooms large would deliver 10,000 rooms immediately. Although some of the early textual worlds did experiment with this kind of setup (MirrorWorld 1986), on the whole, it did not find favor: the resulting rooms were boring and too similar compared to those of the linked-node system. Ease-of-implementation issues were outweighed by the loss of flexibility in design that they implied. What, then, could possibly be gained by switching to a grid format?

The answer is: swift access to a *visual* representation of the world.

Graphics and Grids

The main reason for having textual worlds was that computers in those days had only limited display devices. Although some primitive graphical games were developed at around the same time as *MUD* and were independent of it (most notably *Avatar* (Maggs et al. 1979) on the PLATO system), they never broke free of their hardware and had negligible influence on the future evolution of the virtual world genre.

A more enduring line of graphical virtual worlds began with *Islands of Kesmai*, which was written in 1981 (Islands of Kesmai 1981). Its world was a square grid rendered in ASCII graphics. Character combinations represented different features of the environment, with "[]" meaning a wall, "~~" water, "{}" a tree and so on. In the early 1990s, a number of *IOK*-inspired virtual worlds were written that replaced the ASCII squares with bitmaps; the view presented to the player was from an angle directly overhead, usually with north fixed at the top of the screen. This "flat graphics" approach met with only limited success, however, and the line died out.

The first graphical worlds to overtake their textual forebearers in popularity came from the *MUD* tree. They rendered views in 2.5D – i.e. their worlds were still represented internally in 2D but were displayed as if they were 3D. They adopted an isometric viewpoint, a trend being established by *Ultima Online* (1997), *Nexus: The Kingdom of the Winds* (1996) and *Lineage* (1997). Nevertheless, their internal representation was not all that different from *IOK*'s.

The reasons for the switch from nodes to tiles were entirely based on visual impact. Players could still feel that their characters were in "rooms," but those rooms were now made up of several squares. The granularity had become finer.

To virtual world designers, this tessellated representation presented new challenges:

> The constant scale meant that distance could no longer be finessed. Twenty rooms of textual wilderness meant 200 squares of isometric wilderness.
>
> Because distance changed, so did velocity. Some means of fast transportation was needed so that the time required to travel between distant points was acceptable to players (teleporting, ships, portals, etc.).
>
> The world was 2D. Buildings were restricted to a single story, and caves and bridges were impossible to include.

The latter was particularly problematic. The practical solution was to introduce a degree of nodality back into the system. Parts of the location-definition array were given over to self-contained areas that could not be reached by regular means. Access was gained through particular squares flagged as being *coincident*. If on the main map you entered a square containing a staircase leading upwards, it would teleport you to a submap of the floor "above"; if, on that sub-map, you entered a square containing a staircase leading downwards, you'd be taken back to the main map.

Although the granularity became finer than what was commonly available in a node-based world, it was far from ideal. In particular, anything large and curved was a problem (paths, rivers and circular buildings showed their right angles), and the world *felt* as if it conformed to the squares. When you moved, you moved from the center of one square to the center of another. Walls, roads and coastlines followed the boundaries of squares. Trees, furniture and rocks neatly occupied squares and were arranged in tidy, equidistant patterns. People, animals and monsters followed right-angled lines of movement. You *knew* you were in a world made of squares, which was a little disappointing as that is not quite how the real world is ...

What if the world were not made of squares, but of polygons instead?

Graphics and Polygons

Fully 3D[1] worlds arrived with *EverQuest* (1999), and this is where we are today. Almost all of the several hundred commercial virtual worlds currently in development display their content as a 3D scene. In this method, the one-to-one mapping between

1 There was a 2.5D world, *Meridian 69*, which presented its players with a first-person point of view and therefore looked 3D.

the structure of the virtual space and the software data structure that models it no longer exists. It has been replaced by a relationship in which sets of polygons represent *surfaces*, whether those surfaces are of the ground, of buildings, of denizens or of objects. Space is that apparent volume that does not have a surface in it. Thus rather than creating a set of nodes or squares and saying that each node/square corresponds to a particular location, game designers instead define a location as a mere point in a 3D coordinate system. Most of the world is empty so does not need to be represented explicitly in a data structure; lists of objects within visual range are maintained instead and rendered in terms of the polygons that make up their surfaces.

This approach uses surfaces because that's all that a video card needs to know in order to display an object's image. Objects are, in fact, hollow inside. Indeed, sometimes glitches in the world model or incomplete collision-detection by the camera can mean you get to see what's inside a creature or below a planet's surface.

Although 3D overcomes the main problem of 2.5D in that it allows for things like multistory buildings, its principal advantage is that it looks less blocky; its greater persuasiveness, not its associated representational capabilities, is the main reason why developers prefer it. This isn't to say that modern virtual worlds are full of smooth surfaces and wavy lines – they're not. On close inspection, wheels might be octagonal, for example. Nevertheless, the current virtual worlds are far superior to those that were tile-based and thus locked into having square-edged terrain features no matter what.

Shards and Zones

The three representations of virtual worlds follow a path of increasing detail: from room to space to point. In a 3D world, you are located by a point, but feel you occupy a space, which is bordered by planes to give the impression of a room.

There is also a path of decreasing detail, with groups of related rooms themselves forming conceptual *areas*. Beyond these, there are two even greater abstractions, both of which emerged from implementation issues: shards and zones.

Virtual worlds, much as the real world, can only hold so many people. Their limiting factor is *content* – that which, if players are thought of as consumers, is consumed. Content is a hard concept for non-gamers to grasp, but it's the stuff from which players fashion the events that they find fun. A virtual world with many things that the players want to do has much content; one bereft of desirable activities lacks content. The problem is that a virtual world may have sufficient content for only a certain number of players before those players start treading on each other's toes. As an analogy, consider Disneyland: the park is full of many fun activities, but if it didn't shut its doors when it got full, then few visitors would get to experience them.

Virtual worlds have an advantage over Disneyland in that they can easily be cloned. If there is only enough content for 5,000 people, a copy of the virtual world can be set up so that the overflow has somewhere to go. This is a long-established practice; even *MUD* was able to crank up a new incarnation of itself if it filled up with players. Most commercial virtual worlds today[2] will open with multiple instantiations running on separate sets of computers. Thus even though it's common to refer to the cities and continents of virtual worlds as if they were unique, it should

The main exceptions are *EVE Online* and *Second Life*, which each run only one instantiation that can be scaled up as new players arrive. < 2 be remembered that actually there may be dozens of them. Saying, "I've been to the top of the Tower of Althalaxx" is not the same as saying, "I've been to the top of the Eiffel Tower"; actually, you've only been to the top of *a* Tower of Althalaxx.

These instantiations are commonly termed *servers*, but the term *shard*[3] is also prevalent and is the more correct because it refers to the virtual world itself rather than the hardware upon which it runs. A shard is a single instance of a virtual world.

Zones are different geographical regions of a virtual world that have the same look throughout – a tundra zone looks different than a jungle zone, for example. In textual and 2.5D worlds, this is no big deal, but in 3D worlds, the textures for each zone have to be preloaded into memory by the client software so that they can be displayed as soon as they are needed.

Shards do not interconnect geographically: you can't go east from one shard and find yourself in another. Zones do interconnect and can thus be drawn as a network of nodes (as can areas and, of course, rooms). Although graphical worlds may eschew nodes as a data format, they nevertheless continue to use them as a conceptualizing framework – sometimes for rooms, often for areas and always for zones. These *are* just conceptual nodes, though; the underlying geography of modern virtual worlds still denies designers the flexibility that was possible in textual worlds.

Instances

In graphical worlds, handling crowds is a problem. A textual world can hold arbitrary numbers of players in a single location, but a graphical world has to grapple with certain occupancy problems. What happens when two characters try to stand in the same place? If characters can block each other, a crowd becomes an impassable object. Because of this, the trend is to disable collision-detection for characters in 3D virtual worlds: you can move your avatar completely through the space occupied by another character. This is somewhat fiction-busting, but players have come to accept it as a fact of (virtual) life.

Such a solution has a side effect, though. If multiple people can occupy the same physical space, then they can also simultaneously access the content associated with that space. Textual worlds could fairly easily include alternative content nearby, but this is not possible in graphical worlds as the constraints of visually simulating reality mean there simply isn't room for it. The problem is exacerbated by the higher number of players that circulate graphical worlds compared to textual ones. Basically, 50 people may want to access an area in which there's only content enough for five.

The modern solution is to switch to an *instance*. This takes the coincident squares idea of 2.5D worlds and converts it to that of coincident *planes*. You step through such a plane, and you're transported to a self-contained mini-world beyond it. The idea is extended, however, by allowing multiple copies of the mini-world to exist. One group of players will go to its own, private instantiation of the place – an instantiation that will disappear when it is abandoned, but that can never be accessed by other players. The next group of players will be taken to *its own* replica of the place instead. Thus, instances can be regarded as sharded zones of a virtual world.

3 > The term is *Ultima Online*'s. As fictional cover for why there were multiple canonical copies of the Ultima universe, its designers used the wonderful metaphor of a mirror breaking into a myriad of shards, each one able to reflect what the whole reflected.

In the case of instances, the links between zones again become concrete rather than conceptual. You can enter an instance by going north from a point, but upon immediately going south from within the instance reappear at a different point. Other long-lost designer tricks such as rooms-within-rooms are also made possible by the instance mechanism.

Future Worlds

The way that space is represented in a virtual world is strongly related to how that space is displayed: the closer to reality its appearance, the more constrained its representation. Over time, virtual world designers have developed techniques to address some of these constraints, leading to solutions that are now considered to be part and parcel of the virtual world paradigm.

These solutions reprise the older, flexible and representational structures that were present in textual virtual worlds: a world of nodes (text) has become a world of polygons (3D) arranged as a network of nodes (zones and instances). We're almost back to where we started.

Almost. Actually, we've gained something new: if we can shard instances, we can *also shard zones*. From a designer's point of view, this opens up exciting new possibilities!

Although textual worlds *could* have sharded their nodes, they didn't – their designers didn't give it any thought. Instances, which were introduced to solve a problem that textual worlds didn't have, now make sharding a node fairly routine. Yet nodes don't have to be instances; they could be zones. Suddenly, overlapping, coincident worlds become possible in which players who make one decision are taken to one version of a zone, but players who make a different decision are taken to a different version. From the point of view of the individual player, the world is consistent; from a god's-eye view, it's anything but – a multidimensional layering of differently phased zones, each one personal to every player, but at heart fundamentally the same. In this scenario, actions *could* have global consequences, but only for those involved. Your group might cause a volcano to explode, exposing a strange netherworld of troglodytes and their demonic slaves, but to the players who haven't unleashed the necessary magic, the same volcano is still just a mountain. This is a startling opportunity for designers to do something really *new* with virtual worlds.

What began as a partial solution to the representational problems posed by the higher look-and-feel expectations of players has thus led to an improvement of the original representation itself.

◈ J. Aspnes (1989), *TinyMUD*, Carnegie Mellon University. ◈ R.A. Bartle & R. Trubshaw (1978), *MUD*, University of Essex. ◈ A. Cox (1987), *AberMUD*, University of Wales, Aberystwyth. ◈ P. Curtis (1990), *LambdaMOO*. ◈ *EverQuest* (1999), developed by Verant Interactive, published by Sony Online Entertainment. ◈ *Islands of Kesmai* (1981), developed by K. Flinn & J. Taylor, University of Virginia. ◈ *Lineage* (1997), developed and published by NCSoft. ◈ *Avatar* (1979), B. Maggs, A. Shapira, D. Sides et al., University of Illinois. ◈ *MirrorWorld* Input/Output World of Adventure (1986), developed by T. Rogers, L. Wood, N. Billington run by P. Cordrey ◈ *Nexus: The Kingdom of the Winds* (1996), developed by Nexon Inc., published by KRU Interactive. ◈ K. Nyboe, T. Madsen, H. Staerfeldt, M. Seifert & S. Hammer (1990), *DikuMUD*, Datalogisk Institutved Københavns Universitet, Denmark. ◈ L. Pensjö, (1989), *LPMUD*, University of Gothenburg, Sweden. ◈ *Second Life* (2003), developed and published by Linden Lab. ◈ *Ultima Online* (1997), developed by Origin Systems, published by Electronic Arts. ◈ *World of Warcraft* (2004), developed by Blizzard Entertainment, published by Vivendi.