MicroCosm™ Coordinate Systems and Topology

how to reckon where we are when we're out on the Great Plain
by
Chip Morningstar

Lucasfilm Ltd. Games Division
November 14, 1985

Introduction

This document describes the physical organization of the **MicroCosm™**: its topology and coordinate systems. It also explains how to reckon distances and areas and how to indicate regions of the world's surface.

A note on notation: things in *italics* are new terms being introduced for the first time. The italics mean, ''Pay attention, this is a new term. Don't panic if you don't quite understand it. We are either just now explaining it or will explain it real soon.''

Local Topology

The basic geometric unit of the **MicroCosm** universe is the *region*. A region is an independent, bounded section of the Cartesian plane. Conceptually, the bounds of a region could be any sort of geometric figure. For simplicity however, we will only bound regions by rectangles. This allows us to characterize the area of a region with just two numbers: its *X* and *Y* dimensions. Locations *within* a region can be specified using ordinary Cartesian coordinates. Each of the four edges of a region may be connected to one of the edges of some other region. When an avatar crosses a region's edge he enters the region that the edge was connected to. The topology of these connections can make the world geometry either ordinary or quite bizarre. In particular:

- Two connected edges need not be the same length. When this happens, the entrance point along the edge of a region is linearly interpolated from the exit point along the edge of the region departed from.
- The map resulting from the connectivity of the regions does not have to be consistent with Euclidean geometry. In other words, it need not be possible to lay the regions out on a flat surface in any sensible way. Two regions can be connected as if they were adjacent, even though by all other measures they would be considered distant from each other. Direction does not need to be conserved either. For example, the south edge of one region could connect to east edge of another, rather than to a north edge as one might expect.
- Connections between edges are not necessarily bidirectional. For example, if the west edge of region **A** is connected to the east edge of region **B** it does not inevitably follow that the east edge of region **B** is also connected to the west edge of region **A**. It could connect to some other place entirely. In other words, you may not be able to leave a place by the same door you entered.
- The topology does not have to be stable over time. Just because one region is connected to another today you are not guaranteed that it will be connected the same way tomorrow. New regions can appear, old regions can disappear, regions can move around and connections between them can change.

Even though the topology of the world potentially can be very strange, in practice we will try to minimize deviations from the Euclidean norm. Bizarre topology is difficult for most people to visualize. We want players to be able to develop good practical mental (and physical) maps of their environment. To do this it

is helpful to support the illusion that the world is geometrically consistent and well behaved. Such consistency also enhances the magic and mystery of the places where the topology is irregular. A subtle hand here will keep the players guessing about what is possible, making the universe seem richer and more complex than it actually is.

Coordinates

Within a region, locations are specified using Cartesian coordinates. The origin (location **(0, 0)**) is the region's southwest corner. The X-axis increases to the east and the Y-axis increases to the north. All coordinate values are integers. Distances on the Plain are measured in*feet*, so coordinate pairs gi ve a location's distance in feet from the X- and Y-axes of its region. If we limit the size of regions to a maximum of 255 feet on a side we can pack a coordinate pair into two bytes. This helps speed communications. Even without this limit, we would not want regions to be much bigger because a region is defined to be an area that can be seen in its entirety from a single fixed viewpoint. The one byte ordinate limit still gives us a maximum region area of close to one and a half acres!

Each region has a unique identification number to distinguish it from other regions. These identification numbers are used to tell a player what region his avatar is in and also to express the edge-to-edge mapping between regions. The identification numbers could be the memory addresses of the data structures representing the regions, though we might not want to actually allow such addresses to be known outside the host itself.

Global Topology

The full size of the world has not yet been decided. Suffice to say that it's pretty big, although we don't want the world to be too huge or the players will just rattle around in it. Of course, our flexible region mechanism allows the world to grow or shrink depending on our real estate needs.

Since we don't want to violate Euclidean geometry too severely, we must have a global framework for the various regions to sit in. The basic world is a Cartesian plane mapped onto the surface of a cylinder. See figure 1. The North-South axis runs parallel to the primary axis of the cylinder. At the North and South ends are sheer precipices — the ''edge of the world'' (edges actually). Things thrown off the edge of the world disappear forever. The East-West axis runs around the circumference of the cylinder. It will be possible to circumnavigate the world if you journey far enough to the East or West.

Note that this cylinder is part of the fantasy underlying the **MicroCosm** world. Because of the region-based geometry, there is no functional reason why the world must be this way. We are free to change it if we wish.

Most regions are assigned positions on the cylinder. Such positions are indicated using*global* X- and Y-axes (as distinct from the *local* X- and Y-axes that are relevant within a region). Such positions are merely for our convenience in building the fantasy and have no direct relevance to the underlying implementation. Global positions are ordinals with no units of measure associated with them.

Relative Coordinates

Players (and their home computers) are usually not aware of the actual coordinate system underlying a region. The player systems refer to all locations using a relative coordinate scheme. This scheme uses the same conventions of metric and direction, but the lower left corner of the region as seen from the viewpoint is always used as the origin. All locations are referred to relative to this point. This means that if the viewpoint is on the south side of the region, the player coordinates are the same as the host coordinates. However, if the viewpoint is on some other edge then the player coordinate system is rotated a multiple of 90 degrees about the center of the region with respect to the host coordinate system. Internally, the host keeps track of the viewpoint position and corrects any coordinates generated by the player's system accordingly via a simple calculation.

The relative coordinate system serves the cause of game play by forcing players to move around and explore to find out their orientation with respect to the world as a whole. We may introduce a *compass* object that allows a player to always know his true orientation, but that's a detail.

Distances

As was mentioned above, distances in the **MicroCosm** are reckoned in feet. We won't try to do anything fancy or weird with diagonal distances. We will simply use the good old Pythagorean Theorem, with suitable rounding or truncation of fractions in whatever way makes the host software run fastest. It should be pointed out that ranges (e.g., the firing range property of a projectile weapon object) can be stored internally as distance-squared and thus avoid having to calculate the square root just to determine whether a thing is within the range.

Areas

A given coordinate does not just denote an infinitesimal point on the Plain. Rather, as indicated in figure 1, it also refers to a one-foot by one-foot square area centered around the point. The fact that the point itself is defined as the center of this square, rather than as one of the corners or as something else entirely, is arbitrary. We are doing it this way because it seems simplest, but if some other convention turns out to be more effective in implementation, we are free to change as long as we do so before the first release of the software.