

# Definition of Public Release Habitat<sup>TM</sup> System

by  
*Chip Morningstar*

Lucasfilm Ltd. Games Division  
November 1, 1986

## Introduction

This is a summary document defining the overall *Habitat* system. It describes Commodore 64 software, host software, world design and generation, operations setup, and documentation. The material that follows discusses the intended state of the system in both the beta release version and the public release version.

## Commodore 64 Software

The Commodore 64 software consists of two major components: the program disk and the object disk. Since these have been detailed in previous documents, we will not describe them in much depth here. Briefly, however, the program disk contains the main *Habitat* program, which manages memory, graphics, animation, communications, disk I/O, and the user interface. There are no differences between the beta release and public release versions of the program disk other than bug fixes. The object disk contains the imagery, sound effects definitions, and behavior code for all of the objects that can appear in the *Habitat* world. A complete list of these objects and what they do is given in Appendix A. The only differences between the beta release and public release versions of the object disk are a few objects which are intended to be added during the beta test period. These are indicated in the appendix.

## Host Software

The host software consists of a collection of object behavior routines produced by Lucasfilm that implement the host portion of the behavior of the various objects that can appear in the world and several processes produced by Quantum which manage the running system.

The set of objects implemented on the host matches the set implemented on the Commodore 64, though, of course, the division of functionality between the host and the Commodore means that the two sides of the system do different things.

The primary Quantum-originated processes include the following:

- *Region processor* — responsible for managing communications with and between the players, for controlling execution of the host-resident object behavior code, for providing run-time services required by the host-resident object behavior code, and for implementing *Habitat* functions that require complicated system interface operations. The latter include teleportation, region transition, ESP, and ghosting. The Lucasfilm-originated host behavior code for the objects run as part of the region processor.
- *Database processor* — responsible for transferring information between the *Habitat* disk databases and the region processor's memory.
- *Habitat master process* — responsible for managing the overall operation of the system, coordinating the execution of the other processes and the communications between them.

In addition, the host software includes other ancillary processes, such as the process which initializes new users and all of the various processes that are part of the regular Q-Link system which are required for

*Habitat* to have an environment in which to operate.

There should be no differences between the Quantum-originated host software in the beta release version and the public release version with the possible exception of bug fixes and performance enhancements. The Lucasfilm-originated host object behavior code will be augmented during beta test by the addition of the new objects mentioned previously.

## The World

The habitat world in the beta release version consists of one major city, Populopolis, and surrounding environs. The surrounding area includes sites for the first few adventures as well as a number of social areas and interesting places to simply explore. In addition, for public release we are creating a Habitat Master Geographic Plan describing the locations for new residential areas and new cities to be installed as needed, and the plan for “plugging” new realms into the world as they are created during the normal operation of the system.

For public release, we must generate a world capable of supporting several thousands of players. In order to accomplish this, we are creating a number of automated tools to facilitate world building. These include:

- *Automatic apartment complex generator* — This is a program that will combine a small number of pre-designed regions into the 20,000 turves and supporting hallways, elevators, and so on that we wish to initially generate. This is a simple utility that will be created quickly during the beta test period so that the initial batch of turves can be ready for the public release date. After public release we will port it to host system so that new apartments can be generated on demand as needed.
- *Generalized city generator* — This is a generalization of the apartment generator that will enable the quick generation of areas built out of standard “component” regions according to specifications given in a very simple layout language.
- *Realm editor* — This is actually a family of programs, evolving out of our present development system and leading ultimately to a stand-alone utility for the Commodore 64. The realm editor allows a designer to interactively lay out regions on a map and then to edit the contents and appearance of those regions individually, using the full power of the *Habitat* object set and graphic engine. For the two releases under discussion here we will use it in its present form, an integral part of our current Lucasfilm software development system. As a follow-on project we would like to modify it into a stand-alone program that could be used by third parties.
- *Stochastic region generator* — This is a program that will generate large numbers of variations-on-a-theme regions to flesh out the world for public release. It will be used to create highways, forests, wilderness, and other areas whose regions can be described according to a statistical model.

With the exception of the apartment generator, none of these tools are part of the *Habitat* release system per se. They are simply mechanisms that will enable us to generate the large numbers of regions that will be required to support the public release version of the system.

## The Beta Release World

These are the elements we have either finished or are working on for the beta release *Habitat* world. They are described in greater detail in the earlier document **Coming Attractions**.

### Populopolis

*Populopolis* is the name of *Habitat*'s first town. In its initial (beta release) form, it consists of a downtown area together with residential areas holding turves for up to 320 players. Downtown contains some three dozen businesses to serve the population, as well as a central plaza with the Oracle's fountain in the middle.

The downtown area includes:

- Banks
- News stands

- Bars
- Stores
- A pawnshop
- Theaters
- The Arcade
- Stock Exchange
- Library
- City Hall
- Hall of Records
- Storefronts for rent

### **Other Places**

There are a number of realms for exploration and minor adventures. These are:

- *The Forest Road* — an outdoors realm consisting of a branching road that goes deep into the heart of a mysterious forest.
- *The Really Quite Confusing Desert* — a maze puzzle.
- *The Beach* — a simple social area, designed to look like a beach.
- *The Back Forty* — 40 regions of forest, mostly empty, but home to a pair of mysterious wizards who figure prominently in various *Habitat* intrigues.

### **Things To Do**

In addition to the above geographic areas available for general exploration and treasure hunting, we have also designed a number of more specialized things to do, some of which have special places of their own associated with them. These activities will be coming online progressively during the the beta test period. They include:

- Randy's Region Rally
- D'nalsi Island Adventure
- Capture The Flag
- The Quest for The Tome of Wealth and Fame
- Talk Show
- Game Show
- The Bijou Theater
- Costello's Comedy Club
- Speakers' Corner
- Board Games — chess, checkers, backgammon and othello
- The Game Of Cooties
- The Game Of Machiavelli
- The Habitat Stock Market

### **The Public Release World**

For public release, we will generate turves for 20,000 inhabitants together with associated hallways, streets, and business areas as are required to support that many players. These will be placed, or made ready for placement, according to the Habitat Master Geographic Plan that will also be drafted during the beta test period. In addition, we will add wilderness regions and other adventure realms and activities for these players. The specific nature and number of such regions and activities, however, remains to be determined.

## Operations Setup

A utility to manipulate and monitor the world in operation will be required on the host for the use of both system operators and world generators such as ourselves. The functions required of this utility have been described in the earlier document **Functional Specs for Ghu**. Since the tasks to be performed by this utility require a high degree of host-system interface programming, its development will be Quantum's responsibility. The utility will need to be in operation for public release.

## Documentation

In addition to the various working documents generated during the specification, design, and implementation of the project, the completed set of *Habitat* documentation will include the following:

- *The Official Avatar Handbook* — this is the players' manual. It describes everything that we wish to tell the player about *Habitat*: how to get into and out of the system, how the controls work, how important objects work, and so on. It will be completed for beta release.
- *Guidebook To The Oracle* — this is the system operators' manual. It describes everything related to behind-the-scenes operation of the system: how the system is organized, how to use the operators' utilities, guidelines for acting as Oracle, economic policy, editorial policy, and general rules and guidelines for operations. It will be developed during beta test on the basis of our experiences operating the system. The first draft will be completed for public release, though we expect it to continue to evolve for some time on the basis of operational experience.
- *Habitat Master Geographic Plan* — this is the evolving plan for the expansion of the *Habitat* world. A very rough draft will be completed for beta release, with a "final" draft in time for public release. However, as with the system operators' manual, we do not expect this to be a static document, as the needs of the system will change with system growth and operational experience.
- *Secret Proceedings of The Adventure Masters' Guild* — this is the manual for the "Adventure Masters", the special class of empowered users ranking a level above ordinary players but below the Oracle Itself. This manual will be finished during the initial phases of public operation, as the role and powers of the Adventure Masters are defined.

## Appendix A — Habitat Object Classes

These are the objects to be supported in the public release version of *Habitat*. Those objects tagged with an asterisk (\*) will not be operation at the start of beta test.

amulet — general purpose magic item  
aquarium — turf decoration  
atm — access to bank account  
avatar — represents the player himself  
bag — general purpose container  
ball — throwing around  
bed — turf decoration, seat  
body sprayer — changes color/pattern of avatar  
book — information or amusement for players  
bottle — water for instant objects  
box — general purpose container  
bridge — scenic object  
building — scenic object, access to other regions  
bush — scenic object  
chair — seat  
changomatic — changes appearance of turf  
chest — general purpose container (stationary)  
club — weapon  
coke machine — joke  
compass — direction sensor  
couch — seat  
countertop — object container (stationary)  
crystal ball — oracular frontend  
display case — object container (stationary)  
door — access to other regions  
dropbox — sending mail  
drugs — highly flexible form of magic  
elevator — special intra-building teleport (costs nothing to use)\*  
escape device — teleports avatar back to turf  
fake gun — joke  
fence — scenic object  
flag — scenic object, capture-the-flag prop  
flashlight — portable illumination  
flat — scenic foundation  
floor lamp — stationary illumination  
fortune machine — humor  
fountain — primary oracular frontend  
frisbee — throwing  
game piece — general purpose marker for board games\*  
garbage can — makes objects disappear forever  
gemstone — valuable, general purpose magic object  
ghost — icon for indicating presence of ghosts  
glue — internal use object for scenic purposes  
grenade — weapon\*  
ground — scenic foundation  
gun — weapon  
head — avatar head  
hole — concealable container  
hot tub — social spot  
house cat — joke

instant object pill — mystery object  
key — access to locked doors and containers  
knick knack — decoration  
knife — weapon  
magic immobile — general purpose magic item (stationary), also implements switches and buttons  
magic lamp — oracular frontend (grants wishes)  
magic staff — general purpose magic item, weapon  
magic wand — general purpose magic item  
mailbox — sending and receiving mail  
matchbook — joke  
movie camera — hook for “bugging” regions\*  
paper — user writable text support  
pawn machine — converts objects to money\*  
picture — decoration  
plant — scenic object  
pond — scenic object  
region — internal structuring aid  
rock — scenic object  
safe — lockable container  
security device — exclusion mechanism\*  
sensor — general purpose information collector  
sex changer — switches avatars between male and female  
shovel — opens and closes holes  
sign — conveys information  
short sign — conveys information in fewer bytes  
sky — scenic foundation  
street — scenic object  
streetlamp — scenic object  
stun gun — non-lethal weapon  
table — container (stationary)  
teleport booth — gets avatars from place-to-place quickly  
test — internal object for testing system  
ticket — regulates access to special realms  
tokens — money  
trapezoid — general purpose scenic object  
super trapezoid — very general purpose scenic object  
tree — scenic object  
vendo front — half of vending machine, sells things  
vendo inside — other half of vending machine  
wall — scenic foundation  
window — scenic object  
zone — and internal object used in the realm editor