

Book of WolfMUD



Andrew "Diddymus" Rolfe
Code Crafter's Studio

www.codecrafterstudio.com

WolfMUD Website

www.wolfmud.org

Book of WolfMUD

by Andrew "Diddymus" Rolfe

Copyright © 2002-2003 by Andrew "Diddymus" Rolfe - All rights reserved.

This document was produced in the United Kingdom.

The information is provided "AS IS" without any kind of warranty. The information presented is not a definitive guide to features, performance or commitment to provide any feature. Code Crafter's Studio reserve the right to change specifications and information provided in this document without notice.

The WolfMUD Logo used on the cover is Copyright © Andrew "Diddymus" Rolfe - All Rights Reserved.

Linux is a registered trademark of Linus Torvalds.

Other company and product names may be trademarks or registered trademarks of their respective owners.

Revision History

Revision - 2nd Edition	24th May 2003
Revision - 1st Edition	13th November 2002
Revision - First Public Preview	13th October 2002
Revision - First Private Preview	18th August 2002

Dedication

This work is dedicated to my wife Catherine, daughter Cassidy and my late Grandfather Jack who is sorely missed...

I would also like to thank Sean Geoghegan. Though we have yet to meet face to face, Sean has become a good friend over the last few years. He has provided me with encouragement and been a voice of reason when I needed it the most. Thankyou Sean...

Table of Contents

1. Introduction	1
2. Quick Install	2
Quick Install Procedure	2
3. Player's Guide (Early Draft)	3
What IS WolfMUD?	3
Getting Started	3
Getting Help	3
Your Character	3
Moving Around	5
Items	5
Skills and Guilds	6
Learning skills	6
Nitty Gritty details	6
4. WolfMUD Features	8
5. Requirements	9
Required	9
Optional	9
Tested Environments	9
6. Running Java Programs	11
Java in different environments	11
Running Java on Windows	11
Running Java on Windows using Microsoft's Java	12
Running Java on Unix	12
Running Java on an AS/400	13
7. Installation	14
Java 1.2 and above	14
Java 1.1.x (UNIX)	14
Java 1.1.x (Windows)	14
8. Upgrading from a previous installation	15
Retrospective Upgrade	15
9. Setup and Running	16
Server Configuration	16
Starting the Server	16
MiniWeb Server Setup	17
Binding Ports	17
MiniWeb file serving	17
SETWEB command	17
Setting MIME types	17
Ending The Server	18
WolfMUD Java Client Setup	18
Running the WolfMUD World Builder	19
10. World Builder	20
11. Tour	21
12. Tutorial	26
13. Creating Zones and Locations	27
14. Starting Locations	33
15. Creating Objects	34
16. Creating Mobiles	35
17. Non-Player Barriers	41
18. Linking Zones	42
19. Generating Everything	43
20. Collaboration	44
Exporting and Importing	44
Generated Files Only	44
21. Combat Text	46
Examples	46

22. Compiling from Source	48
Compiling under Windows	48
Compiling under Unix/Linux.....	48
A. Text Dump of the Cabbage Zone	49
B. License Agreement.....	53
Colophon	55

Chapter 1. Introduction

So what is WolfMUD? Wolf stands for World Of Living Fantasy and is a MUD codebase. If you have not come across this type of program before you may already be lost by that remark so I'll start at the beginning...

A MUD is a multi-user dungeon. A text based environment where players can gather and interact with each other and the world around them. You take on the part of a character and through this character explore the world set out before you. You can collect treasures, defeat monsters and build up a set of skills for your character to use and interact with the other players. Depending on the theme of a particular MUD you may find yourself in a medieval fantasy setting, a futuristic science fiction setting or something very bizarre.

A codebase is a program or set of programs that lets you run the MUD world and handles the creatures, locations and objects as well as the player's interacting with everything. The codebase itself is usually taken by the people running the world (Usually called GODS, IMPs (Implementer), Admins, Wizards and there are a lot more names) and they may tweak and modify the codebase so that their world provides a unique experience for their players not found on other MUDs. Typical examples are new Skills, Spells and Races.

WolfMUD provides a starting point for building and running a MUD. If you do not want to modify the code and just create worlds for players you can. If you do want to modify the codebase and add features for your players you are free to do that as well. To help you create your own world a graphical based World Builder is included so that you don't need to be the greatest programmer in the world.

If you can type, you can be creative.

Chapter 2. Quick Install

If you are the type of person who likes to get straight into something without a lot of reading then this chapter is for you. If you get stuck at any point then refer to the chapters that follow for more details on installing WolfMUD, advanced setups, WolfMUD configuration and running Java.

Quick Install Procedure

1. Create an empty directory to install WolfMUD into. Usually this is called `wolfmud`, it will be assumed this is the directory being used.
2. Download the latest WolfMUD JAR files from the WolfMUD web site¹ and save them into the `wolfmud` directory you created. The server JAR file will be called `WolfMUDServer_buildXXX.jar` where XXX represents the build number.
3. Open a shell/command window and change your current directory to be `wolfmud`.
4. Run the JAR file from the command line using:

```
java -jar WolfMUDServer_build303.jar
```
5. Start the WolfMUD server by running the `wolfServer` class.

```
java wolfServer
```
6. Wait for the server to startup and you see the message saying it is accepting clients. Login to the server as a player using:

```
telnet 127.0.0.1 4001
```

Notes

1. <http://www.wolfmud.org>

Chapter 3. Player's Guide (Early Draft)

What IS WolfMUD?

WolfMUD is the 'World Of Living Fantasy' to give it it's full title. It is a game where you create a character that you have control of and send it into the game world to earn experience and fame. While in the game world you will meet other players and creatures that can be interacted with.

Getting Started

When you first connect you will be asked for the name of an existing character, or the name to use for a new character. Enter the name you wish to use for your character. A message will appear saying that the character could not be found and do you want to create a new character. Reply Y. If you are asked for a password the character already exists. Press enter and choose a different name. You will then be asked for a password so nobody else can use your character. You will then be asked to verify your password. Next you get to choose the gender for the character. After you have created your character you will be presented with a menu. If you already have a character you will see this menu after entering your password.

To enter the game take option 1. This enters the main game itself and you will find yourself in a tavern ready to play. While in the tavern you cannot be attacked and cannot attack anything. So it is safe here for you to look at help topics and experiment for a while until you get used to things.

Getting Help

Help is available to all players. To get a list of commands available, use the **COMMANDS** command. If you need help on a specific command use the **HELP** command. It can be used in two ways. **HELP** by itself lists all help topics available. **HELP** followed by a topic will show the help entry for the topic chosen. For example to find out how to use the **BUY** command use:

HELP BUY

You can also ask for help from other players. They may be helpful, then again they may not.

Your Character

If you enter the command **SCORE** (**SC** for short) you will be able to see an overview of your character's main attributes. These are:

- Character
- Date Created
- Last Login
- On Line Time
- Health
- Mana
- Hungry
- Thirsty
- Regen @
- Defense
- Attack

- Combat Damage
- Money
- Encumbrance
- Power Rating

Character Attributes

Character

This tells you your characters name, title, gender and player type. The name and gender you choose for yourself. All players start off as mortals. The players title is based on the highest skill gained. If two skills are of the same level then the first learnt will determine the title.

```
Character...: Diddymus the GrandMaster Mage (Male) Implementer
              \_____/          \_____/      \_____/ \_____/
              Name              Title        Gender  Type
```

Date created

This shows when you created your character.

Last Logon

This shows when you lasted logged onto the server. It will always be when your current session started.

On Line Time

This is the total amount of time you have spent online developing this character.

Health

Health is a measure of how much damage your character can take. The two figures given show your current health and the maximum health. As you take damage the first figure will drop. When it drops to zero your character will die and all equipment and money carried will be dropped where you die. Your health will raise by itself upto the maximum given a little time. However if you are hungry or thirsty the amount of time taken will be longer depending on how hungry or thirsty you are (See also **Regen @**). As your character survives damage over time the maximum health of the character will very slowly rise.

Mana

Mana is a measure of the magical force you have available for casting spells. The two figures given show your current mana and the maximum mana. As you cast spells the first figure will drop. When it drops to zero your character can no longer cast spells until the mana is regained. Your mana will raise by itself upto the maximum given a little time. However if you are hungry or thirsty the amount of time taken will be longer depending on how hungry or thirsty you are (See also **Regen @**). As your character casts more and more spells over time the maximum mana of the character will very slowly rise.

Hungry

This value indicates how hungry you are. As time progresses you will start to get hungry. If you do not eat then it will effect how quickly you regain you health and mana. You cannot die from hunger but if you get too hungry and thirsty you will stop regaining health and mana (See also **Regen @**).

Thirsty

This value indicates how thirsty you are. As time progresses you will start to get thirsty. If you do not drink then it will effect how quickly you regain

you health and mana. You cannot die from thirst but if you get too thirsty and hungry you will stop regaining health and mana (See also **Regen @**).

Regen @

This value shows you the modifier that your health and mana are regenerating at. There are four levels. Full rate (x1.00), half rate (x0.50), quarter rate (x0.25) and no regeneration (x0.00). The current modifier will be a combination of how hungry and thirsty you are.

Defense

Your defense value is an indication of how well you are protected from harm. The value will take into account any armour, magical items and skills etc. that you are using at the time. The higher this value the better protected you are.

Attack

Your attack value is an indication of how well you can attack something. The value will take into account any weapons, magic and skills etc. you are using at the time. The higher this value the better your attack chances.

Combat Damage

Combat damage is a measure of the potential damage you can do to an opponent. It is represented as two figures, for example 5+5. This means your damage will be in the range of 5 points + a random number of points from 1 to 5. So your damage could range from 6 to 10 points.

Money

This is the current amount of money you are carrying. You can have more money than this in an account held by a bank or money changer. Care must be taken as if you are killed you will drop any money you are carrying and may lose it.

Encumbrance

Encumbrance is a measure of how heavy or difficult to carry an object is. It usually works out that 1 encumbrance point is about 1/10th of a pound. The two values shown show how much you are carrying and the maximum you can carry.

Power Rating

Your Power Rating is an indication of how successful your character is in the game world. The value is made up from a number of values including how advanced your skills are and to a lesser extent how rich you are. This determines how much influence you have in the game. Unlike most games of this type running around and killing things does not gain you power!

Moving Around

For each location in the game there will be a number of exits available. To move in a direction type the direction you want to go. Possible directions are **NORTH**, **NORTHEAST**, **EAST**, **SOUTHEAST**, **SOUTH**, **SOUTHWEST**, **WEST**, **NORTHWEST**, **UP** and **DOWN**. You can use **N**, **NE**, **E**, **SE**, **S**, **SW**, **W**, **NW**, **U** and **D** to save typing. Which exits are available will depend on your current location. If you forget what exits are available to you, you can use the command **EXITS** to show just the exits, or use **LOOK** to re-describe the location.

Items

In the game you will come across many useful items. To pick up an item use the **GET** command followed by the item you want to pickup. For example **GET**

SWORD. To drop an item use **DROP** as in **DROP SWORD**. To get more information use the **EXAMINE** command, **EXAMINE SWORD**. You can also try to wield and wear many items. To see what items you currently have and which are worn or wielded, if any, use the **INVENTORY** command (**INV** for short). Many of the commands can also deal with multiple occurrence of an object. For example if there are two swords and you want to pick up the second sword use **GET 2 SWORD**. You can also use the special item **ALL**. As in **GET ALL**, **DROP ALL**, **WEAR ALL** etc.

Skills and Guilds

To help you survive your adventures you can gain useful skills along the way. The most common source of training is from guilds. This will only provide you with basic common skills, you will have to journey and adventure to find the more powerful and rare skills. Training will also cost you more and more the higher the level you wish to train at. Once you have acquired a skill you do not have to pay for more training if you do not want to. Instead by using the skill you can slowly develop it yourself. However this may take many, many repetitions of the skill, more with the higher skill levels.

Learning skills

If you think you have found a potential tutor to learn skills from use the command **LEARN** followed by the tutor you wish to learn from. For example **LEARN WIZARD**. This will show you a list of skills you can learn from this tutor and how much money will be required at your current level. If you have enough money available you can learn the skill if you wish. Some skills require others to be learnt before you can use them. For instance the skill "basic magic" must be learned before you can learn any other magic spells. The fireball spell however cannot be learned until your basic magic is raised to 30%.

Nitty Gritty details

Once you have learned your skills you will have to actually use and practice them so you don't "get rusty". Here is how it works. There are root skills like "basic magic" and "basic combat". Using the **SKILLS** command these skills show as green and have a current and target usage next to them. The current percentage is how much you are using the skill and the target percentage is the minimum percentage you must use the skill. Periodically your skills are checked. If the current usage is below the minimum target usage then the root skill will drop by 0.1% AND SO WILL ALL SUB SKILLS. Sub skills are shown in yellow after each root skill. Using a sub skill will count toward using the root skill as well. For example casting "magic missile" requires you to use your knowledge of "basic magic". This also means as your "magic missile" improves so will your understanding of "basic magic". Due to this for every full 1% a sub skill is raised your root skill will raise by 0.1% as well.

The highest root skill attained will determine your characters title which others will be able to see. If you have learned no skills your title will be "Beginner". The titles currently available are:

- Amateur
- Neophyte
- Initiated
- Skilled
- Practiced
- Proficient
- Studied

- Adept
- Expert
- Master
- GrandMaster

Due to the fact you need to practice skills to keep them there will be very few "power characters" around. For example a GrandMaster Warrior/Mage. The reason for this is that the skills will have to be very carefully balanced with each other. Rely on magic too much and your warrior skills will suffer, fight too much and your magic will suffer.

Chapter 4. WolfMUD Features

Why would someone want to consider using WolfMUD instead of another codebase? Below are a list of some of the features WolfMUD provides:

- Written in 100% Java for use on many different platforms.
- Currently tested on Win95/98/NT, OS/2, Solaris, Linux, RS/6000 and AS/400 but should be compatible with any platform supporting Java.
- Very simple to setup with no compilation needed, just download and run.
- Includes GUI based World Builder allowing creation of worlds without programming.
- GUI World Builder includes features to easily share zones developed by others.
- GUI World Builder tutorial and example world provided.
- Integrated MiniWeb server (with automatic configuration of clients and plugins).
- Stable Core Engine (Max runtime is on Linux at about 986 hours, server ended accidentally).
- Low memory usage (The current world will run in under 1MB with between 10-15 players).¹
- Event driven.
- Scalable - maximum locations, objects and mobile count is around 2,000,000,000 Each!!!
- Large client count depending on platform used (On an AS/400 simulated just over $\frac{3}{4}$ million connections, then ran out of PC power to throw more clients at it).
- Provides a skill based system for players instead of levels.
- Java Developers can easily extend codebase.

Notes

1. This is on top of the memory taken to run the JVM itself which depends on the JVM being used.

Chapter 5. Requirements

This section details the software required to be able to run WolfMUD.

Required

The following are the very minimum requirements on any system for it to be able to run WolfMUD.

- Java Runtime Environment (JRE) 1.1.x or higher (1.3.x recommended) ¹
- TCP/IP Networking (LAN, Dial-up or local)

Optional

The following are optional and allow additional features of WolfMUD to be used.

- Java Enabled Web browser
- A Web Server such as Apache ²
- Swing 1.1 or higher (Only required if you plan to use the World Builder) ³
- Java Development Kit (JDK) for compiling WolfMUD from source

Tested Environments

To date WolfMUD has been tested in the following environments:

- Windows 95/98/NT/2K/ME (Sun, IBM and Microsoft JVMs)
Server, MiniWeb, Client and Builder all run with no problems.
- IBM AS/400 (OS/400 V4R3M0 and above, IBM JVM)
Server and MiniWeb run with no problems. Native telnet can be used as a client with no problems. The AS/400 has no GUI and does not run the Builder or WolfMUD client. Native AS/400 telnet works fine as a client.
- Sun Solaris 7 (Sun JVM)
Server, MiniWeb, Client and Builder all run with no problems.
- i386 Linux (Debian, RedHat, Sun and IBM JVMs)
Server, MiniWeb, Client and Builder all run with no problems.
- PowerPC Linux (Debian, Apple iBook, Blackdown JVM)
Server, MiniWeb, Client and Builder all run with no problems.
- OS/2 (IBM JVM)
Server, MiniWeb, Client and Builder all run with no problems.
- IBM RS/6000 (AIX, IBM JVM)
Server, MiniWeb, Client and Builder all run with no problems.

Note: These only represent the environments currently available to the developers for testing and WolfMUD should run in any environment supporting Java.

Notes

1. For details of the Java Runtime Environment (JRE) and Java Development Kit (JDK) see the JavaSoft web site at <http://www.javasoft.com/j2se> or the IBM web site at <http://www.ibm.com/java/jdk/> where a number of platforms are supported.
2. The Apache web site can be found at www.apache.org. As of Build 240 WolfMUD now has it's own integrated MiniWeb Server.
3. If you are using Java 2 you should have Swing already and not need to download anything extra. If you are using Java 1.1.x you will need to download a 1.1.x version of Swing from <http://www.javasoft.com/products/jfc/download.html>

Chapter 6. Running Java Programs

It is assumed that Java is already installed as is TCP/IP networking. Detailing the setup of Java and TCP/IP networking for all environments is beyond the scope of this documentation. For more details on Java see Chapter 5 which has several references for downloading Java.

There are two types of Java installation. You can install the Java Runtime Environment (JRE) , which allows you to *only* run Java. Or you can install the Java Development Kit (JDK) which allows you to run and also compile and develop in Java. Depending on which environment you have installed the command for running Java may be located in a different directory.

Important: For all environments make sure that the directory you installed WolfMUD into is your current directory before you try running any of the Java classes.

Note for JRE 1.1.x users: If you are using the JRE for version 1.1.x of Java you will need to replace the command **java** with the command **jre** in the following examples.

Java in different environments

Some Examples now follow for running Java in the following environments.

- Windows
- Windows with Microsoft's Java
- Unix
- AS/400

Running Java on Windows

You can check to see if Java is installed and accessible using the **java** command:

```
java -version
```

Depending on the Java vendor and your version it should respond with something like:

```
java version "1.3.0"  
Java(TM) 2 Runtime Environment, Standard Edition (build 1.3.0)  
Classic VM (build 1.3.0, J2RE 1.3.0 IBM build cx130-20000623 (JIT enabled: jitc))
```

If it does not and instead responds with:

The name specified is not recognized as an internal or external command, operable program or batch file.

You will have to specify the full path to the **java** command, as in:

```
c:\jdk1.3\bin\java
```

You can then rerun the above test using:

```
c:\jdk1.3\bin\java -version
```

Now we know the **java** command is working the general command for running Java looks like:

```
java <class file>
```


The entry `<class file>` should be replaced with the the name of the class you wish to run. The class file should be specified without the `.class` extension. For example to run the `wolfServer.class` class you would type:

```
cd c:\wolfmud
java wolfServer
```

Or using a fully qualified path to the **java** command you could use:

```
cd c:\wolfmud
c:\jdk1.3\bin\java wolfServer
```

Important: It is important when running WolfMUD that the WolfMUD directory (the directory you installed WolfMUD into) is your current directory before you try and run any of the WolfMUD classes.

Running Java on Windows using Microsoft's Java

This is very similar to running WolfMUD using a version of Java like Sun's or IBM's (see the Section called *Running Java on Windows*) but the command used should be **jview** and not **java**. Microsoft's Java is usually available if you have Internet Explorer 4 or above. The last point of note: this is a JRE and not a JDK although a Microsoft 'SDK' is available from their website.

Running Java on Unix

You can check to see if Java is installed and accessible using the **java** command:

```
java -version
```

Depending on the Java vendor and your version it should respond with something like:

```
java version "1.3.0"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.3.0)
Classic VM (build 1.3.0, J2RE 1.3.0 IBM build cx130-20000623 (JIT enabled: jitc))
```

If it does not and instead responds with:

```
java: command not found.
```

You will have to specify the full path to the **java** command, as in:

```
/opt/IBMJava2-13/jre/bin/java
```

You can then rerun the above test using:

```
/opt/IBMJava2-13/jre/bin/java -version
```

Now we know the **java** command is working the general command for running Java looks like:

```
java <class file>
```

The entry `<class file>` should be replaced with the the name of the class you wish to run. The class file should be specified without the `.class` extension. For example to run the `wolfServer.class` class you would type:

```
cd /home/diddymus/WolfMUD
java wolfServer
```

Or you could use:

```
cd /home/diddymus/WolfMUD
/opt/IBMJava2-13/jre/bin/java wolfServer
```

Important: It is important when running WolfMUD that the WolfMUD directory (the directory you installed WolfMUD into) is your current directory before you try and run any of the WolfMUD classes.

Running Java on an AS/400

Running Java on an IBM AS/400 is slightly different to normal so help is given here. Firstly it is easiest to run Java from a Unix like shell session or 'QShell'. To start QShell use the command:

STRQSH

Once in the QShell session change to the WolfMUD directory in the Integrated File System (IFS) using:

```
cd \WolfMUD
```

Note: While in the QShell session the commands are processed separate from the session itself, a bit like a batch job. After typing a command wait for the prompt to appear again (usually a \$ or £ sign).

You should now have a QShell session started and be in the WolfMUD directory. To actually run Java classes in this session use the following command:

```
java -opt40 -classpath ./Products/classes <class file>
```

The entry <class file> should be replaced with the the name of the class you wish to run. The class file should be specified without the .class extension. For example to run the wolfServer.class class you would type:

```
java -opt40 -classpath ./Products/classes wolfServer
```

Once the server is running you can use F12 (Disconnect) in the QShell session to detach the Batch server job from the current session. This will free up your session and run the server in batch.

Note: On the AS/400 the access to the IFS can be slow, also optimized versions of the classes will be created so it may take a while for things to happen (many minutes sometimes). *THIS IS NORMAL WHEN STARTING THE SERVER.* This in no way slows down the server once it is running.

Important: It is important when running WolfMUD that the WolfMUD directory (the directory you installed WolfMUD into) is your current directory before you try and run any of the WolfMUD classes.

Chapter 7. Installation

The different parts of WolfMUD are now provided as JAR files and come with automatic installers to make life easier for people. Life is even easier if you are using Java version 1.2 or higher, although the software should run from version 1.1.4 or above allowing use of the Microsoft JVM.

Warning

If you have an existing WolfMUD installation please read Chapter 8 before reading this chapter.

Once the selected JAR has been downloaded place it in the directory you wish the software to be installed into, which may need to be created. Usually the Directory is called `WolfMUD`.

Then depending on the version of Java you are using install using one of the following examples. In the examples the directory specified may be different depending on the location of your Java implementation and the platform you are on (E.G. Windows/Unix). You may also have to fully qualify the location of your `java` command (E.G. `/usr/jdk118/bin/java`). For more information if you are unsure refer to Chapter 6.

Note: Please note, in the examples the upper/lower case is very important even on a Windows platform. Also where example commands are given they should all be typed on a single line even though they may be shown on multiple lines in this document.

Java 1.2 and above

```
java -jar <downloaded JAR name>
```

Example:

```
java -jar WolfMUDServer_build300.jar
```

Java 1.1.x (UNIX)

```
java -classpath .:<downloaded JAR name>:<path to  
classes.zip> install
```

Example:

```
java -classpath .:WolfMUDServer_build300.jar:/usr/jdk118/lib/classes.zip  
install
```

Java 1.1.x (Windows)

```
java -classpath .;<downloaded JAR name>;<path to  
classes.zip> install
```

Example:

```
java -classpath .;WolfMUDServer_build300.jar;c:\jdk118\lib\classes.zip  
install
```

Chapter 8. Upgrading from a previous installation

If you already have a WolfMUD installation and wish to use existing player, project or data files follow these instructions.

1. Backup the important files from you existing installation:
 - a. Copy the `players` directory containing `.ply` files to a safe place (E.G. a temporary directory)
 - b. Copy the `projects` directory containing `.wmp` and `.wmx` files to a safe place (E.G. a temporary directory)
 - c. Copy the `data` directory containing `.zon`, `.loc`, `.mob` and `.tng` files to a safe place (E.G. a temporary directory)
2. Delete the existing installation (E.G. Everything in `C:\WolfMUD` or `/home/<user>/WolfMUD`)
3. Copy the old `players`, `projects` and `data` directories back (E.G. into `C:\WolfMUD` or `/home/<user>/WolfMUD`)
4. Install the new version of WolfMUD using the normal installation instructions in Chapter 7.

Retrospective Upgrade

If you have already installed the new version of WolfMUD and wish to upgrade some player files or project files (E.G. from a friend or previous install):

1. Copy the files you wish to retrospectively upgrade:
 - a. Copy player files into the `players` directory with the rest of the `.ply` files.
 - b. Copy projects and extract files into the `projects` directory with the rest of the `.wmp` and `.wmx` files.

2. To upgrade players rerun the SERVER installation using the **-nounpack** option. For example:

Java 1.2 and above:

```
java -jar WolfMUDServer_build300.jar -nounpack
```

Java 1.1.x:

```
java -classpath .;WolfMUDServer_build300.jar;<path to classes.zip> installServer -nounpack
```

3. To upgrade projects rerun the BUILDER installation using the **-nounpack** option. For example:

Java 1.2 and above:

```
java -jar WolfMUDBuilder_build62.jar -nounpack
```

Java 1.1.x:

```
java -classpath .;WolfMUDBuilder_build62.jar;<path to classes.zip> install -nounpack
```

Chapter 9. Setup and Running

It is assumed you have already installed WolfMUD before reading this, if not you probably need to read Chapter 7 on installation first.

Server Configuration

There are a number of options that can be configured for the WolfMUD server however for most people the defaults should be fine to start with until you get adventurous.

In the directory WolfMUD was installed into there should be a file called `server.properties` in the `config` directory. This file can be edited and the default values changed. Each property and it's values are fully documented in the file itself. Two properties of particular interest will be **server.welcome** and **server.news** by changing the default values you can create a custom welcome screen and news items for your MUD.

If you just want to override a value temporarily without changing the `server.properties` file certain properties can be specified on the command line when starting the server. Specifying properties in this way will override settings in the properties file only until the server is ended. A full list of options can be seen by starting the server and appending `-?` to the end of the command. As an example:

Java 1.2 and above:

```
java wolfServer -?
```

Java 1.1.x:

```
java -classpath .;c:\jdk1.1.7\lib\classes.zip wolfServer  
-?
```

This should display the following details:

Valid parameters:

```
-?          This help screen  
-port <port> Port to listen on for clients.  
             Where <port> is the port number (1-65535)  
             NOTE: 1-1023 usually reserved for other services.  
-nodisklog   Do not write out a log to disk.  
-noconsole   Do not log to a console/screen.  
             Useful if running in as background process.  
-noeventlog  Do not include events in the log.  
-maxClients <x> Maximum number of client connections allowed.  
             If not specified defaults to platform limit.  
-timeout <x>  Timeout limit for idle players where x is a  
             number of minutes. Defaults to 15 minutes.  
-noweb       Do not start internal web server.  
-webport <port> Port for web server. See -port option.
```

Starting the Server

To start the server you need to run the `wolfServer` class (See Chapter 6 on running Java if you need help). This will start the server running. You can optionally specify a number of parameters to override some settings in the `server.properties` file (See the Section called *Server Configuration* for details).

By default the Client Port which players should connect to is 4001 and the Web server is started on 4002. The window / process the server is running in should now be left alone to do it's job. It will not end until the server is ended.

Note: The FIRST character that is created will be created as an Implementer and has access to the **#SHUTDOWN** command to end the server.

MiniWeb Server Setup

If you do not want to use the MiniWeb Server it can be disabled using the **-noweb** command line option. You can also change the port the web server is running on by using the **-webport** option or by editing the `server.properties` file. To save having to type the port number just for web pages, For Example `http://127.0.0.1:4002`, you can run on the web server on the default HTTP port of 80. However on some systems this may require special system security configuration as port numbers in the range 1-1024 are usually reserved for well known services and access may be restricted to these ports on your server.

The last line displayed by the server when it starts up should look like:

```
27May 16:47:12.35 [M] Web server waiting on port 4002 (Accepting on any IP Ad-
dress)
```

So you should be able to open a web browser and goto **http://<my_IP>:4002** where <my_IP> is the IP address of the WolfMUD server EG `http://123.045.78.9:4002`. If you cannot see this line, because the server is running as a background process for example, *ON THE SERVER MACHINE* goto `http://127.0.0.1:4002/statusPlugin.wgp`. This will let you check the ports being used.

Binding Ports

In the above section when the server starts up the last line displayed is:

```
27May 16:47:12.35 [M] Web server waiting on port 4002 (Accepting on any IP Ad-
dress)
```

Notice it says (Accepting on any IP Address). This indicates that any available IP address on the server machine will be used. Normally this is fine. However if your machine is multi-homed (E.G. has more than one network card installed or is configured for multiple IP addresses) you may want to specify a specific IP which players can connect to. This is done by setting the **server.bindIP** property in the `server.properties` file.

Note: This binds the Telnet port and web server port.

MiniWeb file serving

In the installation directory there is a directory called `web`. This is the only directory (and subdirectories) that the MiniWeb Server will allow access to. Files in other directories cannot be requested by browsers. At the moment HTML pages cannot use URLs that start with a period, For Example `..` for previous directory, as it opens up a security hole. This will be addressed in a later release.

SETWEB command

As of build 258 the SETWEB command has been removed. The setup of the MiniWeb server is now totally automatic as is the setting up of the WolfMUD Client.

Setting MIME types

When a request for data is made by a browser the MiniWeb Server will send the requested file (if applicable) and will also try to tell the browser what type of data the file should contain. For example if a `.txt` file is sent the browser will be told it is of type `text/plain` and the browser should know what to do with that file type. The mapping between the file extension `.txt` and the MIME Type `text/plain` is configured in the file `mime.properties`. This can be found in the `config` directory.

Ending The Server

To end the server you need to log on as a character that is an Implementer and use the **#SHUTDOWN** command. This will end the server in an orderly manner and save any characters currently in use. If the server is just killed off there is a chance that the socket is not released and the server will not restart because it thinks the socket is in use. Sometimes the server will restart, but players will not be able to connect. Also any characters in use will not be saved and the character will revert back to when it was last auto-saved. This could lead to some very unhappy players.

WolfMUD Java Client Setup

There are three ways to setup the client: Automatically, as a stand-alone application or as an Applet on a Web server. The following sections detail each of these methods. For quick setups and testing method 1 is recommended. Also this method allows a number of people to quickly get playing on a network or over the Internet. Method 2 is of use if the internal web server is not being used or needs to be configured. This method is also handy if your browser does not support Java or you want to use a different version of Java then your browser supports. Method 3 is usually used when you already have a web server setup or just don't like the integrated MiniWeb server.

AS/400 Users: The Client will NOT run on the AS/400 in a normal "green screen" session, only the server will. It is possible to access the server using Telnet from a "green screen" session.

- Method 1: Automatic setup

This is the quickest and easiest method to use. When the MiniWeb server is started it will automatically configure itself and the client pages. Get the players to point a Java enabled web-browser at the server and start playing. (See the section on MiniWeb Server Setup for more details on the Port to Point too).

- Method 2: WolfMUD Client stand-alone Application setup

In the distribution file there is a file called `wolfMUDClient.zip`. This file should be unpacked on each machine you wish players to connect from, usually into a `wolfMUD` directory. To start the client run the class `wolf.client.wolfClient`. This class can also have 1 or 2 parameters. The first parameter is the host to connect to. It can be specified as the Host Name or the Host IP address. The second parameter is the port number to connect to. The port can only be specified if a host has been specified. The defaults are 127.0.0.1 for the host and port 4001. To specify the parameters just type them after the class name. For example:

Java 1.2 and above:

```
java wolf.client.wolfClient 127.0.0.1 4001
```

Java 1.1.x:

```
java -classpath .;c:\jdk1.1.7\lib\classes.zip
wolf.client.wolfClient 127.0.0.1 4001
```

Note: The commands should be entered on a single line. For more information on running Java classes see the chapter Running Java if you need help.

- Method 3: Stand-alone Web server setup

To setup the client so it can be accessed from a stand-alone web server the web server and WolfMUD server must be on the same machine. This is due to restrictions that the Java *'sandbox'* security model imposes to protect users from Applets loaded over the Web accessing their own machine. Also the integrated web server must not be running on the port you want to use.

If you want to use a stand-alone web server AND want to make use of the automatic configuration and plugins available from the MiniWeb server you can just embed hyperlinks to the MiniWeb plugins into your pages. In the installation there is a directory called web. This contains an HTML file which contains such links and you can crib from these.

Warning

The client and server files should not be put into the same directory, otherwise players will be able to get to the server code and classes as well and download the whole lot from your server. The WolfMUD server does not have to be installed anywhere near the server files as long as they are on the same machine.

Running the WolfMUD World Builder

To start the World Builder:

1. Change to the directory where the software is installed (Usually WolfMUD).
2. Run the class `wolfBuilder`

- a. For Java 1.2 and above:

```
java wolfBuilder
```

- b. For Java 1.1.x make sure the Swing classes are in you classpath:

```
java -classpath .;c:\jdk1.1.7\lib\classes.zip;
c:\swing-1.1.1fcs\swingall.jar wolfBuilder
```

Note: The commands should be entered on a single line. For more details on Swing for Java 1.1.x see the chapter on Requirements for details. For more information on running Java classes see the chapter Running Java if you need help.

Details on using the World Builder including a tutorial are covered starting with Chapter 10.

Chapter 10. World Builder

This section details how to use the World Builder to create your own worlds with locations, mobiles and objects without having to do any programming.

When the Builder is loaded the Builder workbench will be displayed with the World Tree frame opened as shown in Figure 10-1.

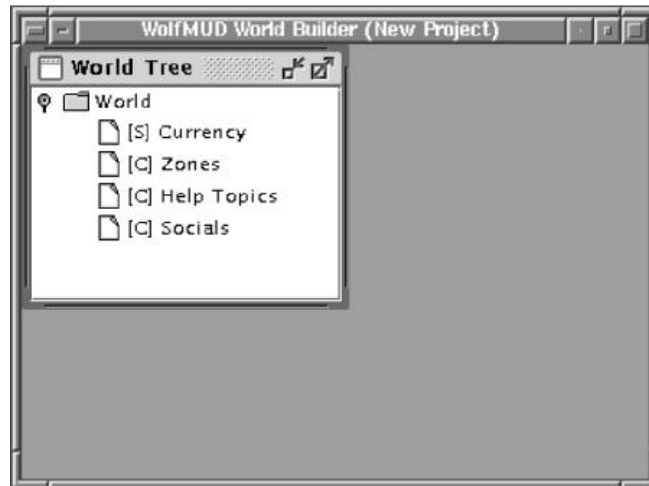


Figure 10-1. World Builder Workbench

Most actions are performed by using a right mouse click (or mouse click with a modifier key). To make sure everything has loaded properly: select the Zones entry in the World Tree and right click. A menu will be shown. From this menu select **Add**. A submenu should be shown with a single item on it: Zone. This is shown in Figure 10-2. If the entry for Zone is not displayed close down the builder and try starting it again (This bug was fixed and should no longer be a problem).



Figure 10-2. World Tree Showing Zone

Chapter 11. Tour

Included with the World Builder is a sample project file based on the Stock world shipped with WolfMUD. To load it select **World** from the World Tree, right click and select **Load** as shown in Figure 11-1.



Figure 11-1. Loading a Project

This will display a project load dialog from which you can select different projects to be loaded into the Workbench. The sample project is called WolfMUDWorld.wmp. Select it and click the **Load** button as shown in Figure 11-2.

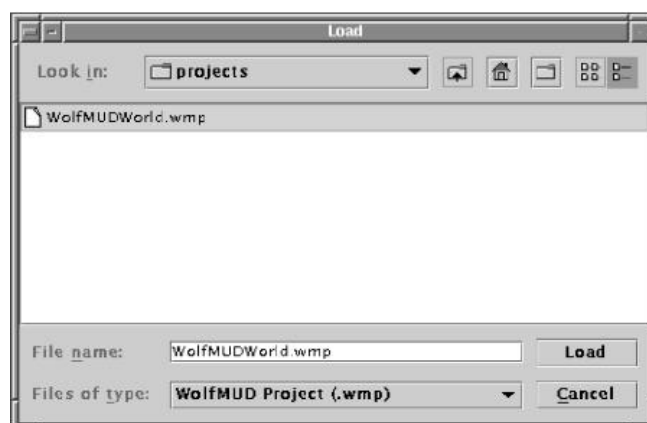


Figure 11-2. Project Load Dialog

You will now see a progress indicator as the project is loaded, and then as the World Tree is rebuilt. See Figure 11-3.

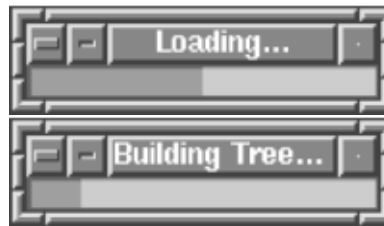


Figure 11-3. Load and Build progress Indicators

When the project has loaded and the World Tree has been rebuilt expand the **Zones** by double clicking the tree entry or selecting the icon left of the entry. Under the **Zones** tree entry there should be 3 zones available. If you expand each zone you will see there are three more entries available: Locations, Mobiles and Objects as shown in Figure 11-4. Notice under the zone "South of Zinara" the icons for **Mobiles** and **Objects** look different and have no "handles" next to them. This is because no mobiles or objects have been defined for this zone.

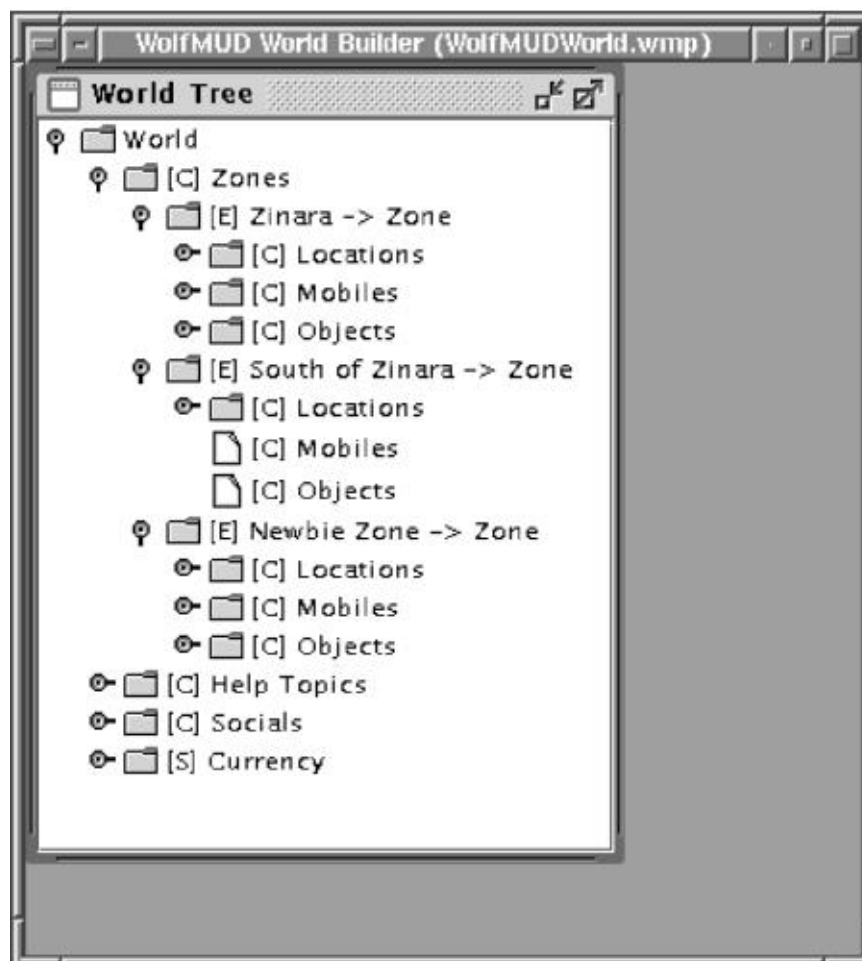


Figure 11-4. Zones Expanded

You will notice looking at the tree that next to each entry is a letter in square brackets. There are three possible values [C], [S] and [E] indicating the type of entry. See the Table for an explanation of each type.

Tree Mnemonics

[C] Multiple Category

This type of entry cannot be edited but unlimited additional entries can be made under this entry. For example you can add many creatures under the **Mobiles** entry.

[S] Single Category

This is similar to a [C] type entry but only a single entry can be made under it. For example when creating a mobile there is a Single Category Entry for the mobile's body so you can only add one body per mobile.

[E] Entity

An entity entry is the actual entry in the tree that holds the data for an object in the game. Individual creatures and objects are examples.

If you find the entry for the **Newbie Zone** and expand the entries for **Mobiles** You will see a number of creatures that have been defined for this zone as shown in Figure 11-5.

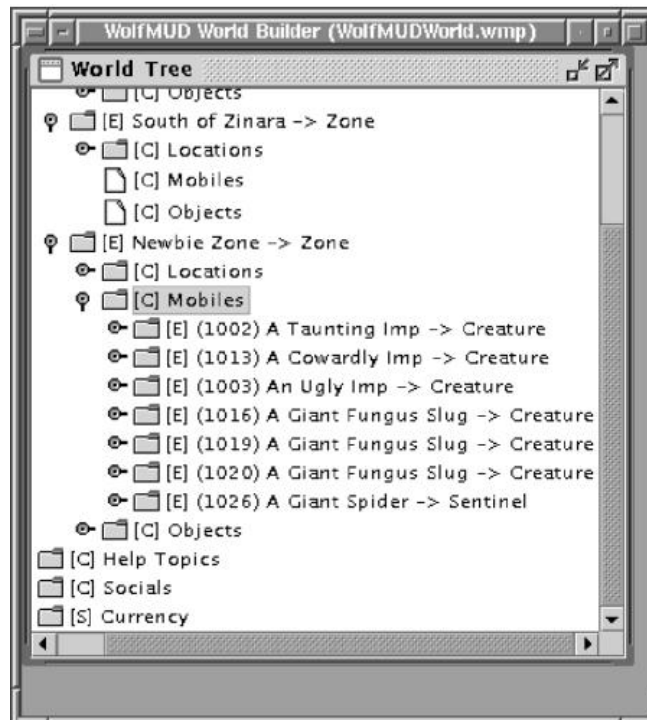


Figure 11-5. Mobiles for the "Newbie Zone"

For any tree entry with a type of [E] you can right click and display a popup menu. From the World Tree find select the entity for **An Ugly Imp**. Right click to bring up the popup menu and select **Edit...** as shown in Figure 11-6.

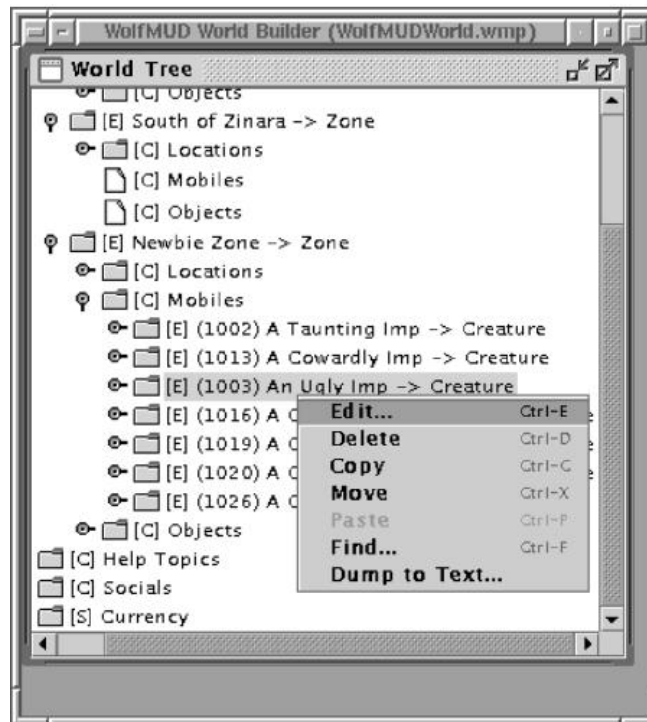


Figure 11-6. Selecting an entry for editing

A new frame will open showing the properties defined for the Ugly Imp. Tool tips are available for each attribute and there is also a main help panel for all types available. See Figure 11-7 for an example.

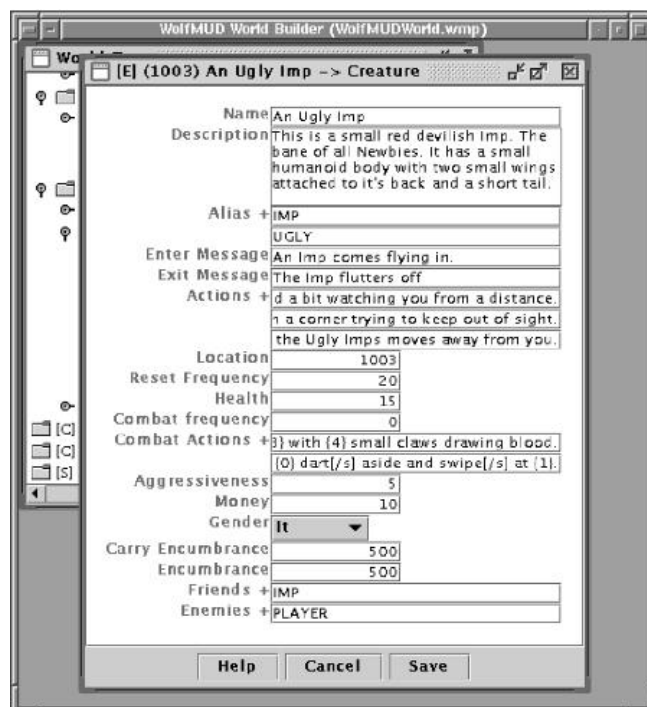


Figure 11-7. Attributes for an Ugly Imp

If you look back to the World Tree you will also see a number of items under the tree entry for the Ugly Imp. These include the Imps body, Inventory and Attributes. Figure 11-8 shows these elements. Each element with an [E] can be edited and if you look at the Pouch you will notice it is a container and actually contains something.

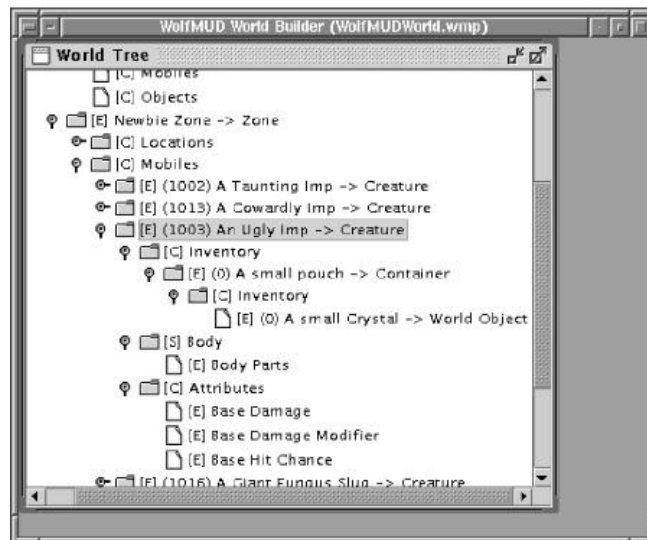


Figure 11-8. Additional Attributes of the Ugly Imp

There is also a find feature available which will look for any text that has been entered. Select the **World** entry in the World Tree and right click to display the popup menu. From here select **Find...** A new frame will open. In the field for **Search Text** enter the word **water** and click the search button. After a short pause any result will be shown. Each item in the result list can be double clicked and this will position you to the entry in World Tree. Figure 11-9 shows the find dialog in action.

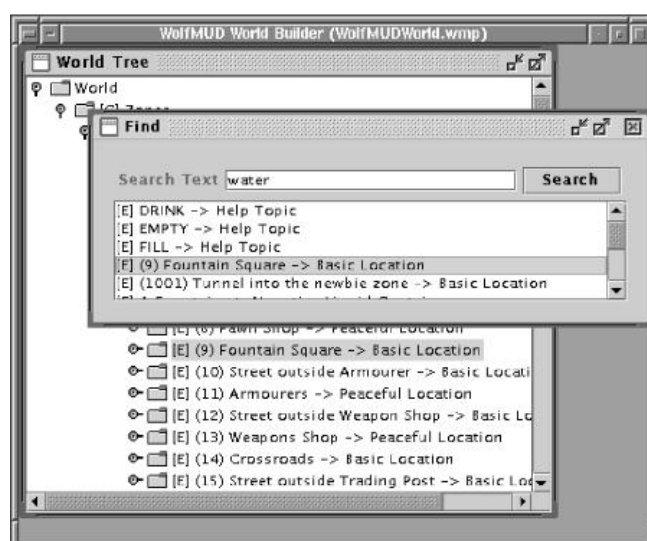


Figure 11-9. Find Dialog in action

Chapter 12. Tutorial

An example nows follows that will show you how to:

- Create a Zone
- Create Locations
- Create Mobiles
- Create Objects

The zone to be created will be added to the zone South of Zinara. Figure 12-1 shows a map of this zone with the new zone to be created shown in blue. It will be called The Cabbage Zone.

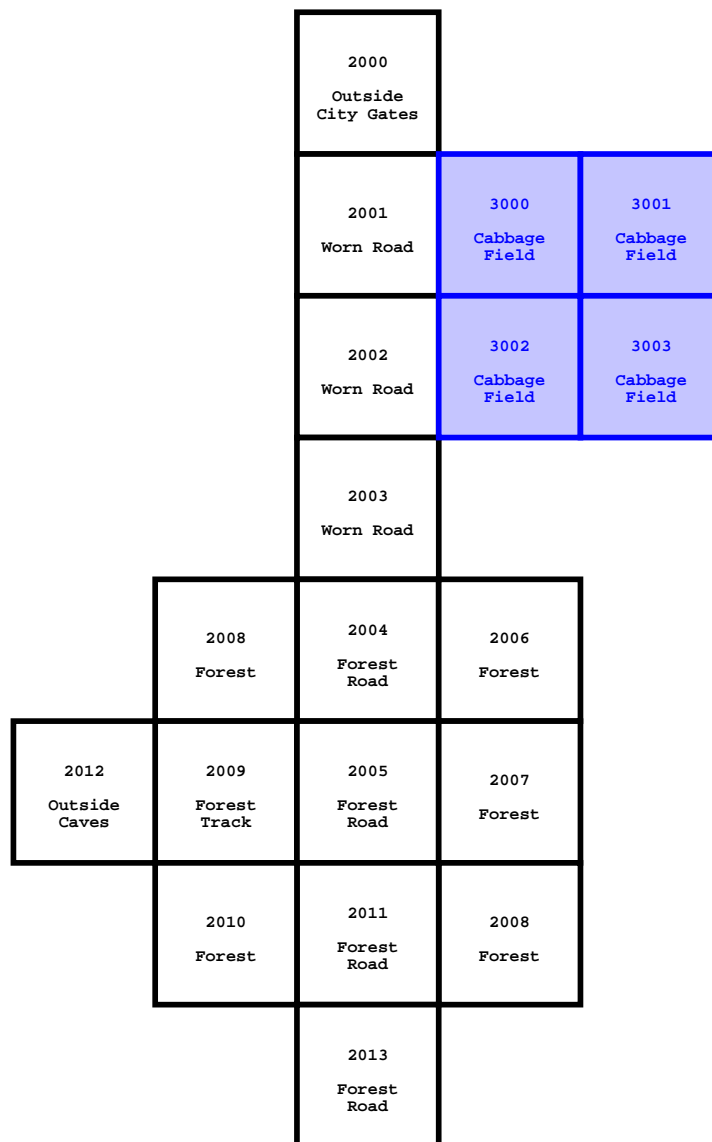


Figure 12-1. The Cabbage Zone (In blue).

Chapter 13. Creating Zones and Locations

From the World Tree select the entry for **Zones**, display the popup menu and select **Add**. A sub menu should popup where **Zone** can be selected. See Figure 13-1.

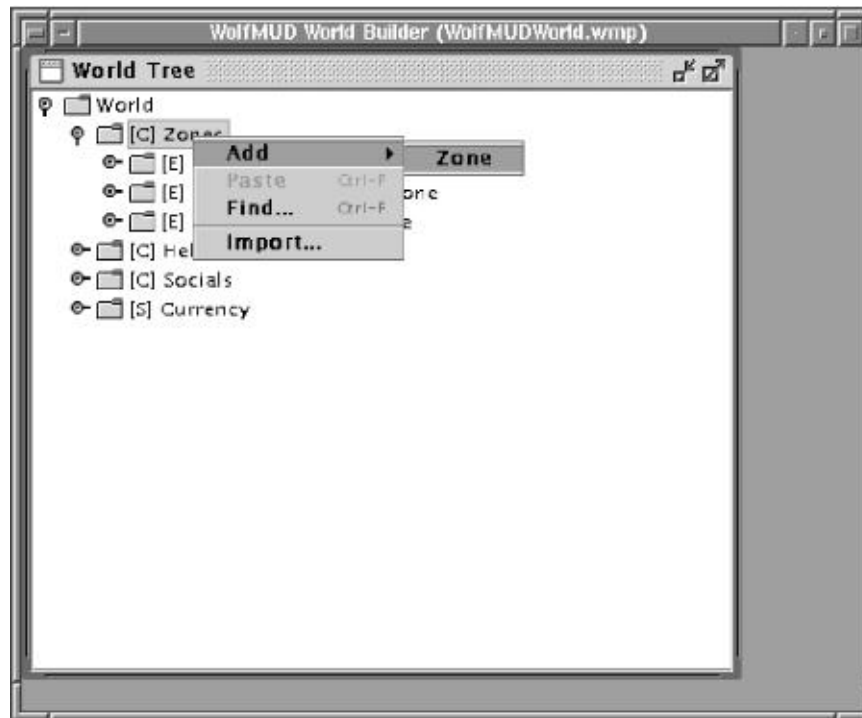


Figure 13-1. Adding a new Zone

Selecting this will open a properties dialog for the new zone details. Enter the details as shown in Figure 13-2 and then select the **Save** button.

Note: When creating a new zone you must fill in the Alias. This is what the generated files will be prefixed with.

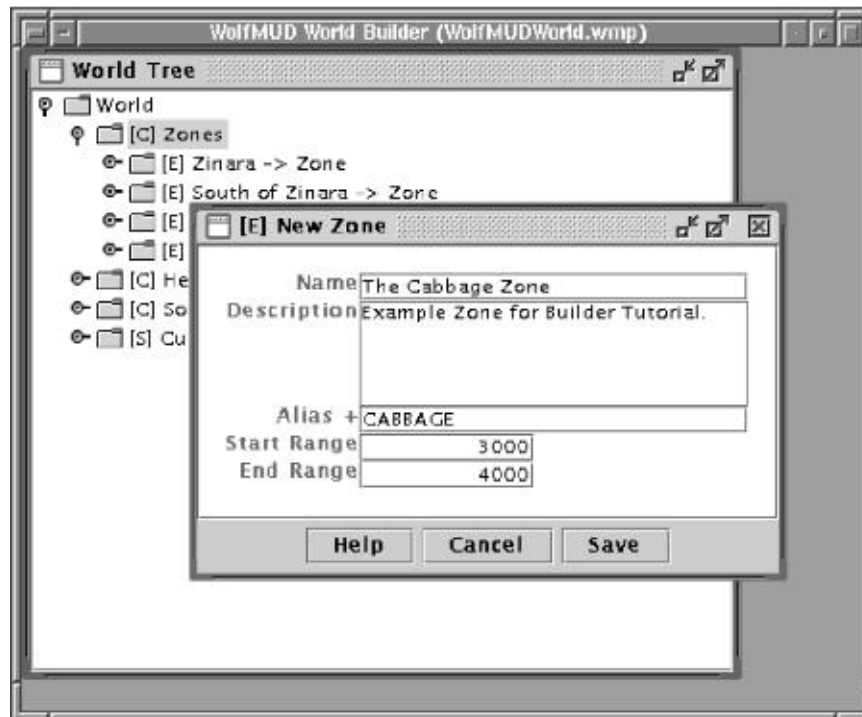


Figure 13-2. Properties for the Cabbage Zone

There will now be a new zone listed in the World Tree; Our Cabbage Zone. Select and expand the entries under it. You should see Locations, Mobiles and Objects. Select **Locations** and display the popup menu. The select **Add**, this will show a sub menu of different location types you can have. For now select the **Basic Location** type. Figure 13-3 shows an example.



Figure 13-3. Adding a Location

This will now cause a new properties frame to be opened, this time for entering the details for a Basic Location. Enter the details as show in Figure 13-4.

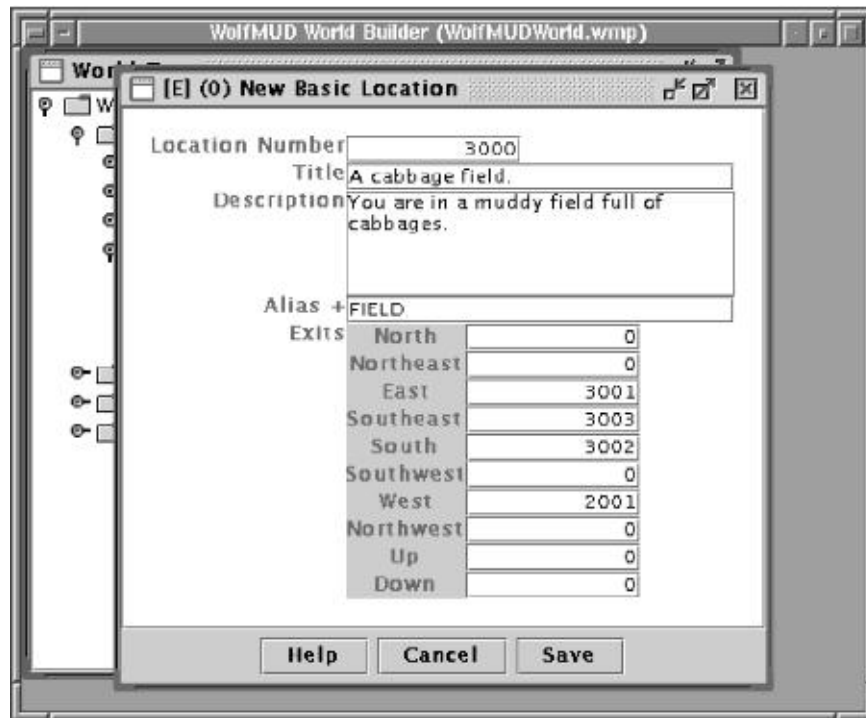


Figure 13-4. Properties for a Basic Location

Now if there are cabbages in the field the players should be able to interact with them, but we don't want them to take them. To do this we add a narrative object to the location. Select the new location and expand it so the **Narrative Objects** category can be seen. Select this category and display the popup menu, select **Add** and from the type of narrative available select **Narrative Object**. See Figure 13-5.

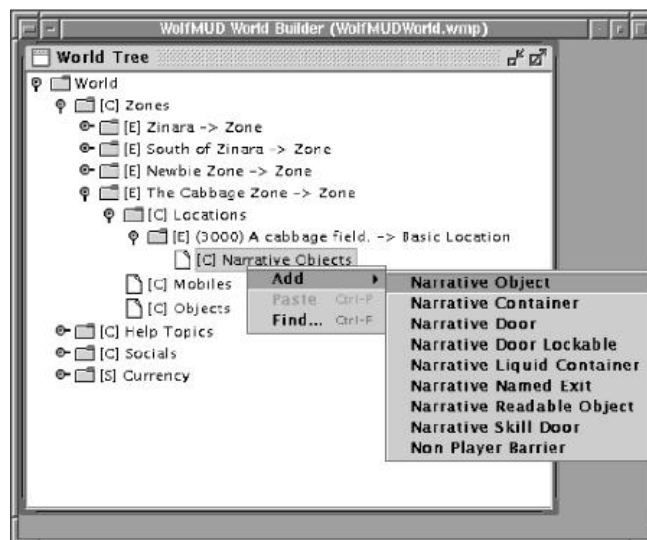


Figure 13-5. Adding a Narrative Object

Again we get a new dialog to enter detail for the Narrative Object. This is where details for the cabbages can be entered as shown in Figure 13-6. Now a player may type '**examine cabbage**' or '**examine cabbages**' so we need both aliases. To get the field for the second alias select the first field and right click to display the popup menu, select **add**. A new field will now be added. If there is more than one field already a delete option will also be available.

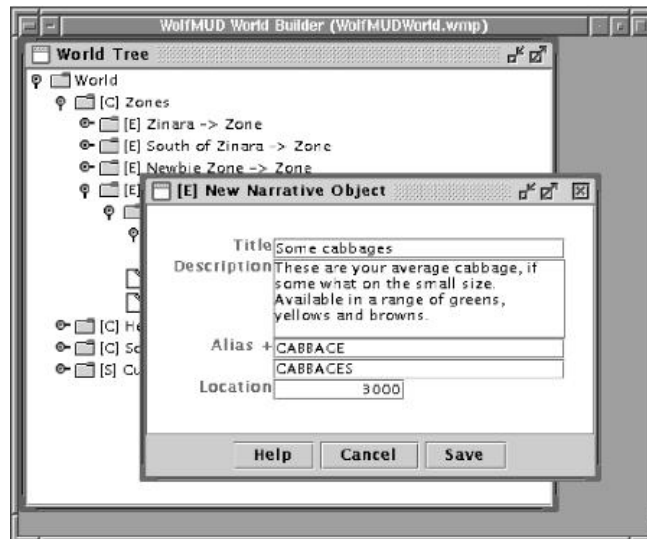


Figure 13-6. Properties for a Narrative Object

So there is now a field with cabbages we can examine. There are another three fields to do each with cabbages, but we don't want to have to type everything in again. Select the location and from the popup select **Copy**. Display the popup again select **Paste**. Display the popup twice more and paste each time. You should now have four locations in total. Notice that the narrative under the locations has also been copied for you. See Figure 13-7.

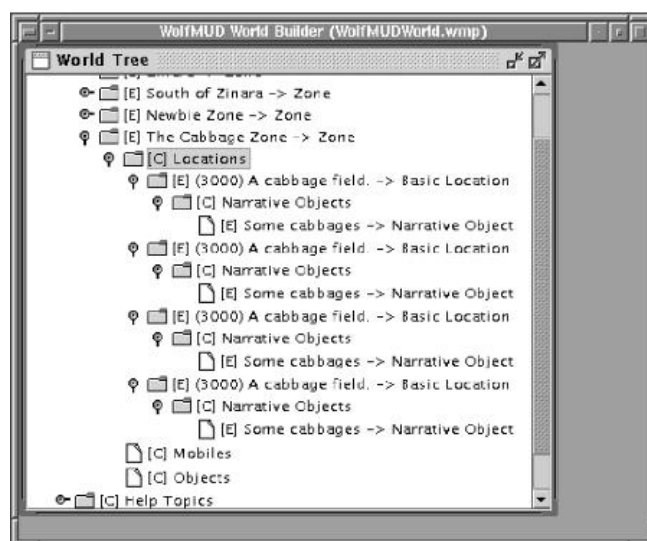


Figure 13-7. Copied Locations and Narratives

The three new locations and narrative are exact copies of the original so you need to edit each of them. The information that needs changing is the location index

and exits for the locations and the location index for the narratives. Figure 13-8 shows the modified locations and Figure 13-9 shows the modified narratives.

The figure shows three screenshots of the WolfMUD World Builder (tutorial.wmp) interface, each displaying a location configuration window for a cabbage field. The windows are titled "[E] (3001) A cabbage field. -> Basic Location", "[E] (3002) A cabbage field. -> Basic Location", and "[E] (3003) A cabbage field. -> Basic Location". Each window contains the following fields:

- Location Number:** 3001, 3002, and 3003 respectively.
- Title:** A cabbage field.
- Description:** You are in a muddy field full of cabbages.
- Alias:** FIELD
- Exits:** A table with 12 rows (North, Northeast, East, Southeast, South, Southwest, West, Northwest, Up, Down) and 2 columns (Exit Name, Exit Value).

The Exit values for each location are as follows:

Exit Name	Location 3001	Location 3002	Location 3003
North	0	3000	3001
Northeast	0	3001	0
East	0	3003	0
Southeast	0	0	0
South	3003	0	0
Southwest	3002	0	0
West	3000	2002	3002
Northwest	0	0	3000
Up	0	0	0
Down	0	0	0

Figure 13-8. Modified Locations



Figure 13-9. Modified Narratives

Chapter 14. Starting Locations

Starting Locations in WolfMUD are a special type of Peaceful Location. They have all the properties of a Peaceful location, that is no combat is allowed providing a safe location. However starting locations also provide points of entry into the world for players. Creating a Starting Location and its properties are exactly as for creating a Basic or Peaceful location.

Starting locations provide a way to implement “home zones” for players. If a zone has at least one starting location defined then that zone becomes a “home zone”. When a player creates a character (or uses an existing character without a “home zone” set) they may be asked to choose a “home zone”. Figure 14-1 shows an example login.

```
WolfMUD (C)1999,2000

World    Created By
Of       A.Rolfe (AKA Diddymus)
Living
Fantasy

Please enter your characters name or the name for a new character.
>Diddymus
Please wait, checking for character...
A character by that name cannot be found. Create a new Character? (Y/N)
>y
Please enter a password for your character.
>password
Please enter your password again for verification.
>password
What gender would you like your character to be:
    M - Male
    F - Female
>m
1st Character Created - Player Type is IMP.
Please select where in the world you would like to start from:
    1. South of Zinara
    2. Zinara
>
```

Figure 14-1. Example Login

If there is only one “home zone” available this will be automatically selected for the character and no option to choose will be shown. However if there are more “home zones” the player is asked to choose one.

If a “home zone” has more than one starting location defined then each time the player logs in one of them will be selected at random when the player enters the world.

Chapter 15. Creating Objects

Next we will add a prize cabbage a player is able to take from one of the fields (Location 3000). Find the tree entry for **Objects** under the **Cabbage Zone** and select it. Then display the popup menu and select **Add** a sub menu will be displayed showing all the different types of objects that can be created. Select **Food** from the sub menu. See Figure 15-1.

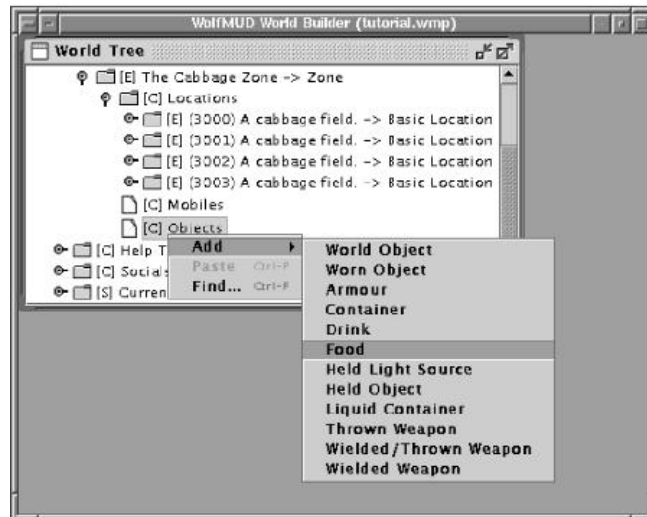


Figure 15-1. Adding a Food Object

This will cause a new editor window to opened so the properties can be edited. Enter the properties as shown in figure 21 to create the prize cabbage.

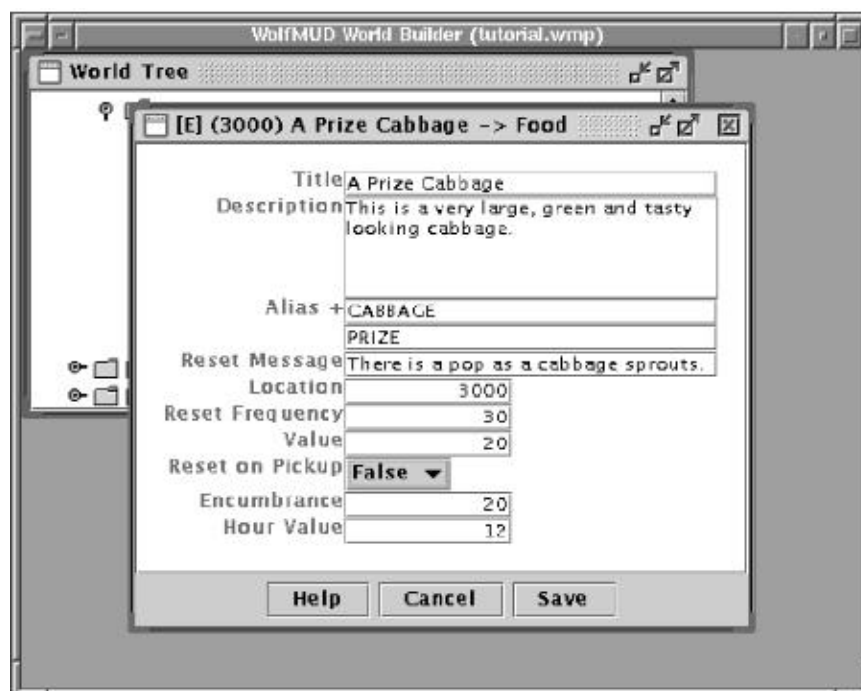


Figure 15-2. Prize Cabbage Details

Chapter 16. Creating Mobiles

If we have fields of cabbages then maybe there should be some to tend them. For this we will create a Cabbage Farmer (Location 3003) to toil in the field. Find the tree entry for **Mobiles** under the Cabbage Zone and select it. Then display the popup menu and select **Add** this time a sub menu will be displayed showing all the different types of mobiles that can be created. Select **Creature** from the sub menu. See Figure 16-1.

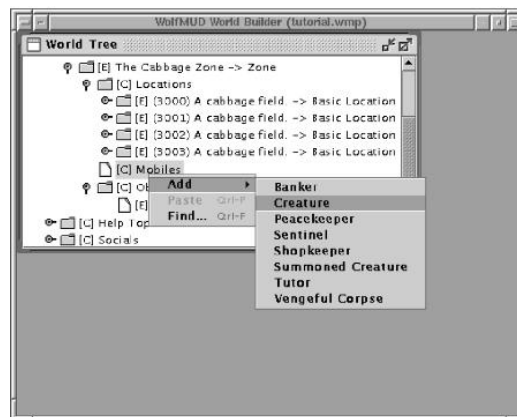


Figure 16-1. Adding a Mobile

This will cause a new editor window to open this time to enter details for the farmer. Enter the properties as shown in Figure 16-2 to create the Cabbage Farmer. The text for the combat actions may look a bit odd due to special formatting. For information on how to compose these messages see the section on Combat Text.

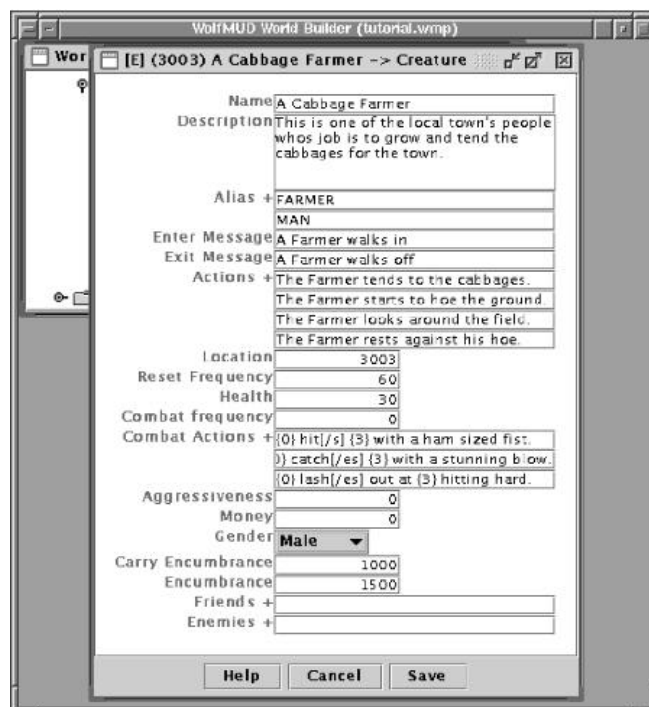


Figure 16-2. Cabbage Farmer Details

Now we have our cabbage farmer we need to flesh him out a bit and add items, a body and attributes. These are added to the categories listed under the entry for the farmer (**Inventory**, **Body** and **Attributes**). The body will be added first. Notice that the category for **Body** has a type of [S] next to it. This means we can only add a single item to this category. Select **Body**, display the popup menu and select **Add**, from the sub menu select **Body Parts**. See Figure 16-3.

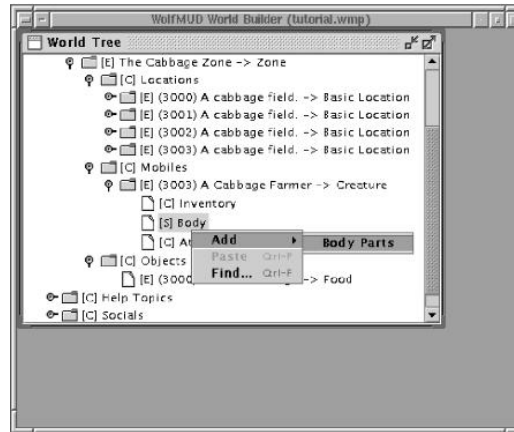


Figure 16-3. Adding a Body

This will show a frame where the body parts can be entered. The body parts a mobile has will effect what items can be worn/held/wielded. For example if a creature has no hands, for example a giant slug, it cannot use any item that requires hands to be used. Our cabbage farmer will have a standard humanoid body as shown in Figure 16-4. Notice how easy it is to create mobiles with a hand/arm/leg or even an ear missing.

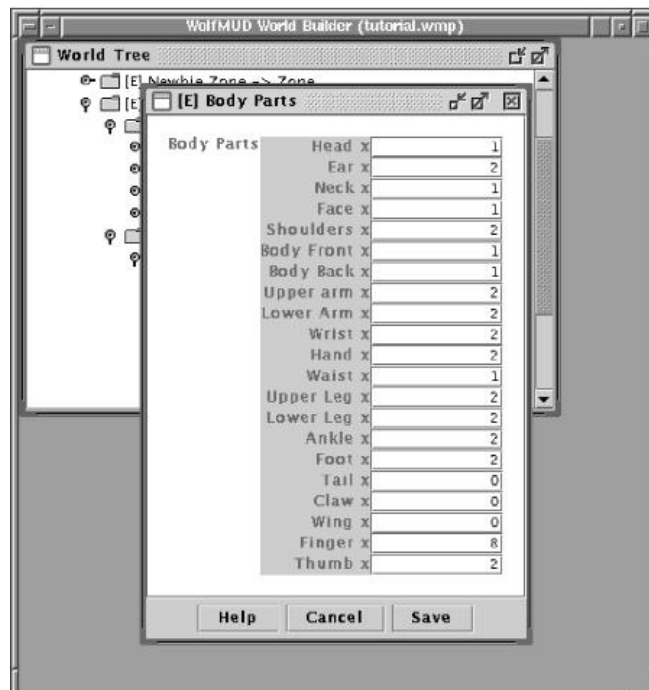


Figure 16-4. Body Details

Next add the attribute details. Select the **Attributes** category for the farmer and add one each of **Base Damage**, **Base Damage Modifier** and **Base Hit Chance**. The values for each are shown in Figure 16-5.

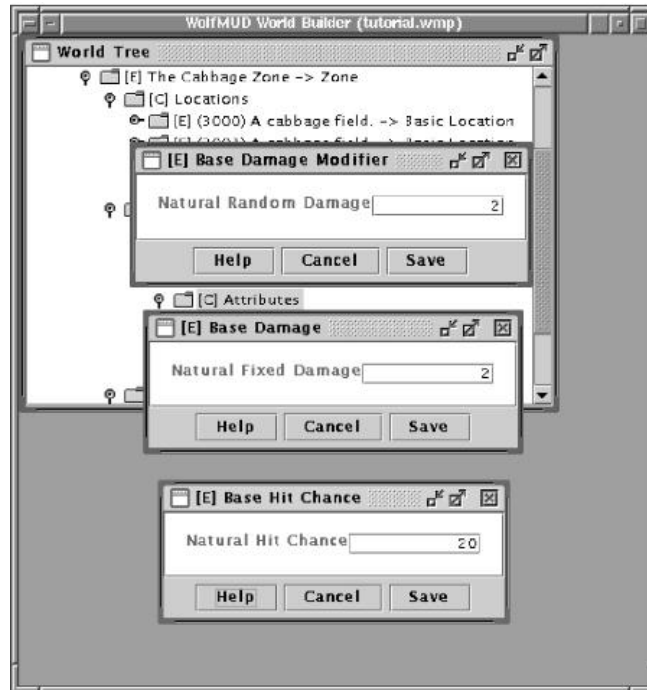


Figure 16-5. Attribute Details

Lastly we need to add some items for our farmer. Figure 16-6 shows the items in the Farmer's inventory. He has clothing A Rough Shirt, Some Leather Trousers and A Pair of Leather Boots (Figure 16-8). He also has a Small Sack with some lunch in it; Some Bread and Some Cheese (Figure 16-9). Lastly he has a Hoe which can be wielded as a weapon should he be attacked (Figure 16-7).

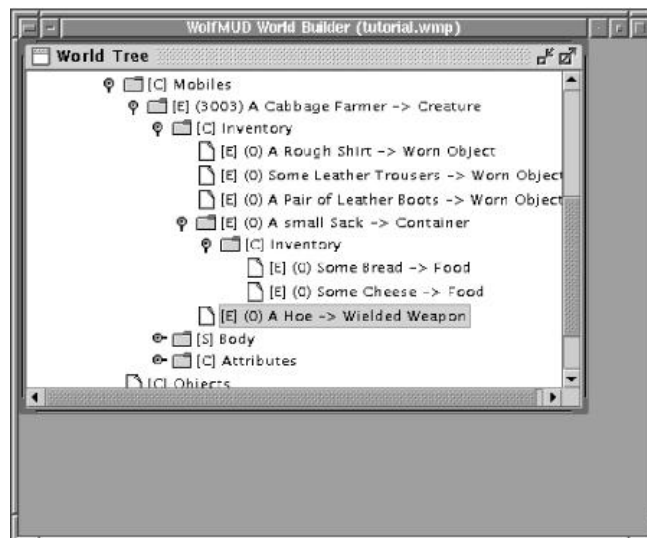


Figure 16-6. Cabbage Farmer's Inventory

Each of these items can be added in the usual way by selecting **Inventory** from the cabbage farmer and adding the different types. The exception to this is the Small Sack. This has it's own inventory. You need to select the Small Sack and use **Add** to add the food into it. Figure 16-8, Figure 16-9 and Figure 16-7 show the attributes for the different items. The text for the combat actions for the Hoe may look a bit odd due to special formatting. For information on how to compose these messages see Chapter 21.

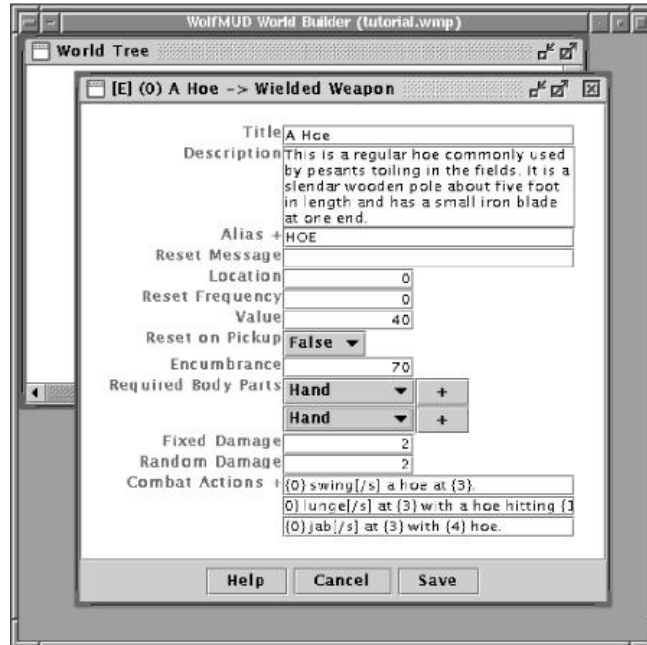


Figure 16-7. Farmer's Hoe

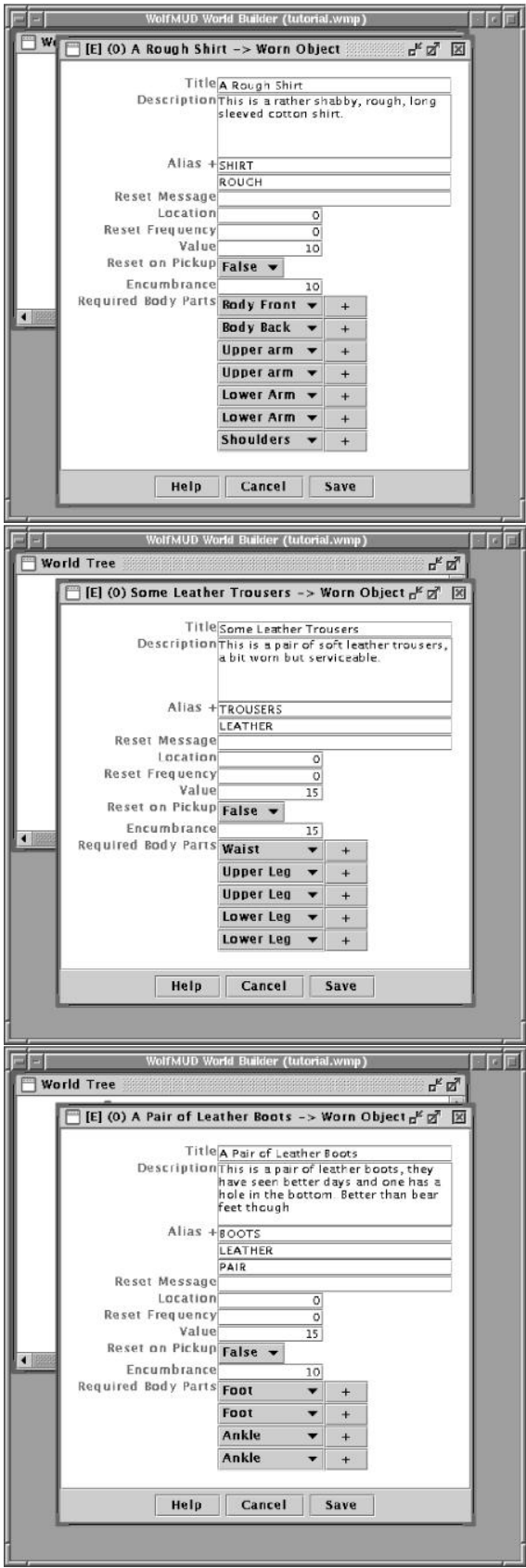


Figure 16-8. Farmer’s Clothing

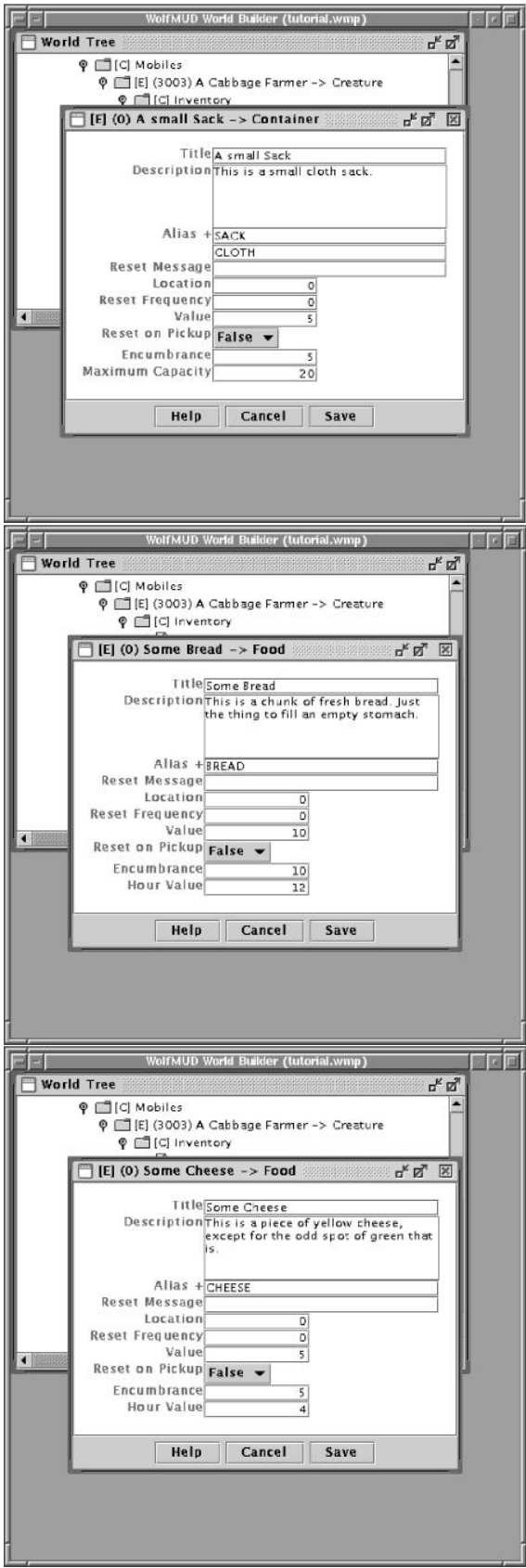


Figure 16-9. Farmer's Lunch

Chapter 17. Non-Player Barriers

Before we generate our zone we need to stop our farmer from wandering out of the fields. This is done using Non-player Barriers. We shall create one at the west exit of location 3000 and one at the west exit of location 3002. Find the **Locations** category for the Cabbage zone. For locations 3000 and 3002 add a **Non-Player Barrier**. The properties for each are shown in Figure 17-1.



Figure 17-1. Non-Player Barrier Details

Chapter 18. Linking Zones

The last thing we need to do before our zone is ready is to link it to the existing zone South of Zinara at locations 2001 and 2002. To do this find the zone South of Zinara, and under it find the **Locations** entry. Under this look for locations '(2001) Worn road south of Zinara' and '(2002) Worn road'. The exits for each of these locations need to be modified and the descriptions changed to indicate the new Cabbage fields to the east of the road. Figure 18-1 shows the changes for each location.

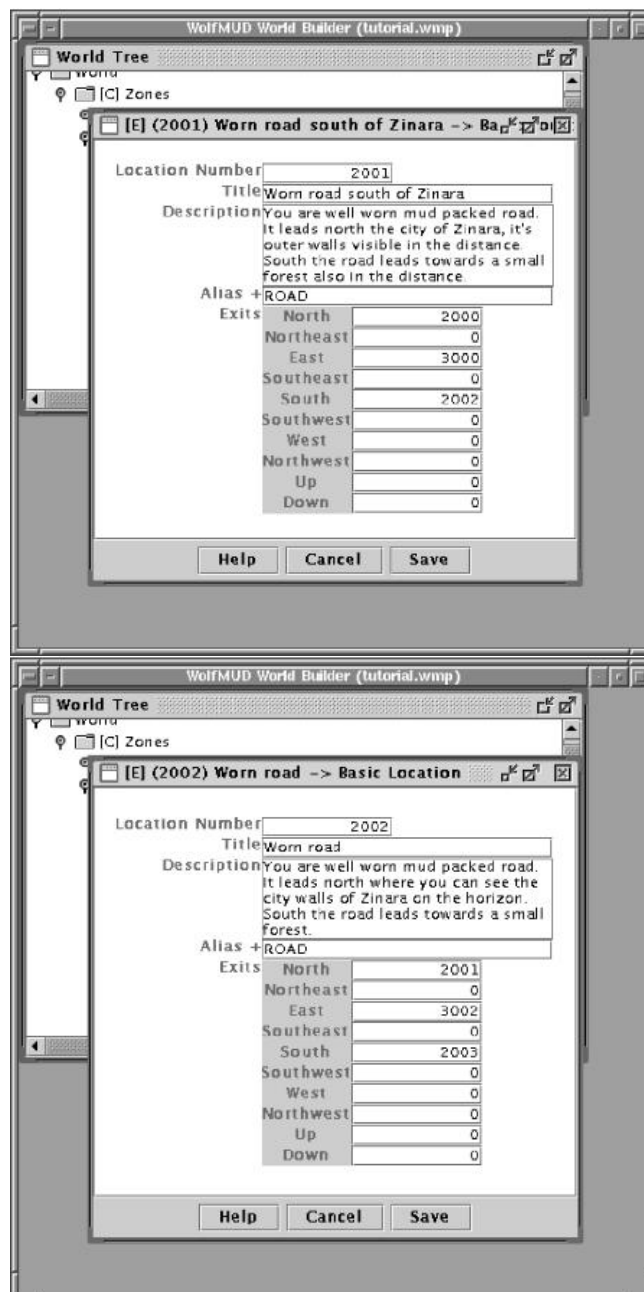


Figure 18-1. 'south of zinara' Zone Changes

Chapter 19. Generating Everything

Now at last the Cabbage Zone is ready to be tried out, but first we need to generate the new zone. Select the entry **World** in the world tree, display the popup menu and select **Generate**. A small progress dialog will open while everything is being generated as shown in Figure 19-1. The yellow bars indicate the progress for a zone and are reset as each zone is generated. The blue bars show the overall zone progress and also any help entries.

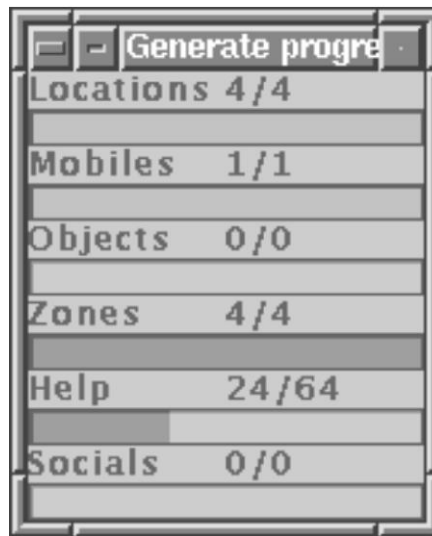


Figure 19-1. The Generate progress Dialog

Now all that is left to do is start the server running and connect to it. Exit the tavern and head south out of town. If everything is working properly the cabbage zone should be just to the east of the main road.

If you can log on as an Implementer: There are a few things you can do as an Implementer that make checking your new zone easier.

- Use the command **DUMP Z #** this will list all loaded zones so you can check your zone is loaded.
- Use the command **GOTO O PRIZE** will take you straight to the `Prize Cabbage` object.
- Use the command **GOTO L L3000** will take you straight to the first room in the `Cabbage Zone`.

Chapter 20. Collaboration

Now you have your new Cabbage Zone there is a chance (slim) that you may want to share your work with others. To allow collaboration between designers working on a world two features have been included. You can either export your zone, or provide just the generated files. When you export a zone to a *.wmx file (WolfMud Export) anyone you give the file to can import it into their World Tree, look through it and modify it. If you give them just the generated files they can only use them 'as is' and you will have to make any changes for them.

Exporting and Importing

To export a zone select the zone to export, display the popup menu and select **Export**. A dialog will be displayed where you can specify the file name for the export. By default this file will be saved into the `projects` directory. You can now take a copy of the *.wmx file and give it to other designers to use. To import a *.wmx file select the **Zones** entry in the world tree. Then display the popup menu and select **Import**. This will display a dialog listing the files available for import. Select a file and the details will be incorporated into you current project.

Generated Files Only

If you wish to keep control over the modifications made to your zone, or you don't want anyone to look at the zone details and spoil any 'surprises' you can do that as well. When a zone is initially created an alias **MUST BE PROVIDED**. This is the name used for the generated files. The generated files will be placed into the `data` directory. In the tutorial we gave the alias of `CABBAGE` to our zone. Therefore once we have generated our zone we should have a number of files prefixed `CABBAGE`. The exact files generated will depend on the details you create. Table 20-1 shows the generated file types.

Table 20-1. Generated File Types

*.zon	Zone Header File This file should always be generated for all zones.
*.loc	Locations File This file contains details about all locations in the zone as well as narrative objects. It is only generated when there are locations.
*.mob	Mobile File This file contains all of the details for each of the mobiles in a zone. It is only generated when there are mobiles.
*.tng	Thing File This file contains details for everything else that can be created except help topics. It is only generated if needed.

To distribute your zone this way just provide copies of the files associated with your zone. To use these type of files on your own server just copy them into

the `data` directory and they will be picked up automatically when the server is started.

Note: If the `*.zon` file is not present then NONE of the other files will be used either.

Chapter 21. Combat Text

Combat text in WolfMUD is a little tricky until you get used to it. Basically when a combat message is displayed it can be seen by one of three types of people: the attacker, the defender and anyone watching (Viewers). This means the same message cannot be sent to everyone. For example if you have the message:

A Farmer hits out at you with a ham sized fist.

This would look fine for the defender but not for the attacker or viewers. If you are watching and saw 'A Farmer hits out at you with a ham sized fist' you would think the farmer is attacking you. Likewise if you are attacking and saw it you would think 'but I'm not a farmer'.

To get around this substitution values can be used to automatically put the right text in the right place for the person it is intended for. The substitution values take the form {number}. See Table 21-1 for a list of valid values.

Table 21-1. Substitution Values

Substitution	Replaced By		
	Attacker's Viewpoint	Defender's Viewpoint	Viewer's Viewpoint
{1}	HIM, HER or IT _{DB}	YOU	HIM, HER or IT _{DB}
{2}	Defender's Name + 'S	YOUR	Defender's Name + 'S
{3}	Defender's Name	YOU	Defender's Name
{4}	YOUR	HIS, HER or IT'S _{AB}	HIS, HER or IT'S _{AB}
{5}	YOU	HIM, HER or IT _{AB}	HIM, HER or IT _{AB}
{6}	HIS, HER or IT'S _{DB}	YOUR	HIS, HER or IT'S _{DB}
{7}	YOU	HE, SHE or IT _{AB}	HE, SHE or IT _{AB}
DB = Defender Based, AB = Attacker Based			

Another problem is that of *tenses*. For example you hit and he hits. To get around this you can use the special form [text/text]. The text before the slash is only seen by the attacker and the text after the slash is only seen by defenders and viewers.

Examples

Situation: A Farmer fighting Diddymus

{0} hit[/s] out at {3} with a ham sized fist.

Attacker sees: You hit out at Diddymus with a ham sized fist.

Defender sees: A Farmer hits out at You with a ham sized fist.

Viewers see: You see A Farmer hits out at Diddymus with a ham sized fist.

Situation: A City Guard (Female) fighting A Taunting Imp (It)

{3} make[s/] a grab for {0} but {7} dart[/s] aside and swipe[/s] at {1}.

Attacker sees: A Taunting Imp makes a grab for You but You dart aside and swipe at It.

Defender sees: You make a grab for A City Guard but She darts aside and swipes at You.

Situation: A City Guard (Female) fighting A Taunting Imp (It)

Viewers see: You see A Taunting Imp make a grab for A City Guard but She darts aside and swipes at It.

To make it easier to compose these messages a small utility is available. In the directory where the server and World Builder software is installed there should be a file called `testActionText.html`. This can be run in a browser or using `appletViewer` from the JDK. Figure 21-1 shows the utility.

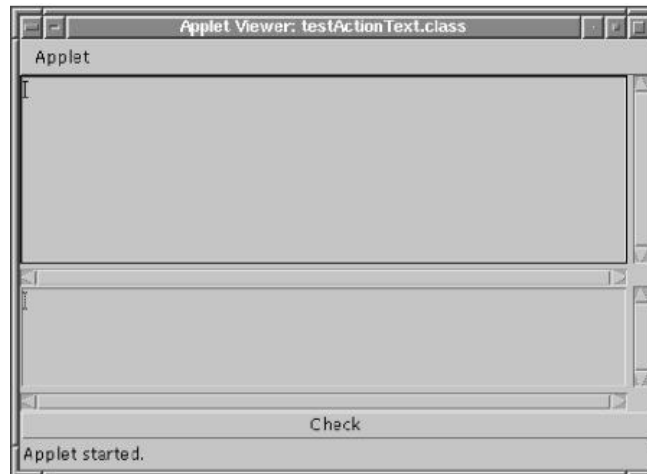


Figure 21-1. Message Utility

The utility assumes a City Guard (female) fighting a Taunting Imp (It). Enter your message in the top half and select the button **CHECK**. The message will be formatted for the attacker [A], defender [D] and viewer [V]. The resulting messages will then be displayed in the lower half for you to check. Figure 21-2 shows an example.

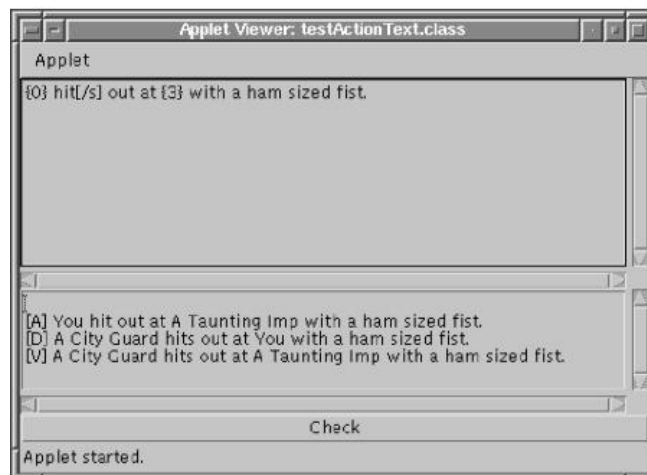


Figure 21-2. Message Utility Example

When you are happy with your message it can then be cut and pasted into the World Builder.

Chapter 22. Compiling from Source

Compiling WolfMUD from source can be very easy even if you are only using the standard Java Development Kit can not a fancy IDE. However just compiling the sources using **javac** on the main `wolfServer.java` will not work and you will find only about a third of the sources are compiled.

You might ask why is this? Compiling the base `wolfServer.java` file causes it and also any classes it depends on to be compiled. However WolfMUD does not directly reference all of its classes, it uses reflection quite a bit. This means the compiler cannot know about the other classes as they are loaded at runtime.

The easiest way to get around this is to quickly check and compile all of the source files automatically. However you may find you need a JDK version 1.3.0 or higher to use the **javac @<filename>** method of compiling as other versions of the JDK have problems especially under Windows.

Compiling under Windows

These commands will build a list of all of the java files which can then be used as input into the **javac** command to compile everything. The commands can either be entered again and again or put into a batch file. It is assumed that WolfMUD has been installed into `c:\wolfmud` and the sources into `c:\wolfmud\sources`.

```
cd c:\wolfmud\sources
dir /B /S *.java file.list
javac -d c:\wolfmud @file.list
```

Compiling under Unix/Linux

These commands will build a list of all of the java files which can then be used as input into the **javac** command to compile everything. The commands can either be entered again and again or put into a script file. It is assumed that WolfMUD has been installed into `~/wolfmud` and the sources into `~/wolfmud/sources`.

```
cd ~/wolfmud/sources
find . -name '*.java' > file.list
javac -d ~/wolfmud @file.list
```

Appendix A. Text Dump of the Cabbage Zone

Figure A-1 show all of the details for the Cabbage Zone. It was produced using the **Dump To Text** menu option on the Cabbage Zone and may be of help if some of the images in the tutorial are difficult to read.

```
The Cabbage Zone -> Zone =====

Name: The Cabbage Zone
Description: Example Zone for Builder Tutorial.
Alias +: CABBAGE
Start Range: 3000
End Range: 4000

(3000) A cabbage field. -> Basic Location =====

Location Number: 3000
Title: A cabbage field.
Description: You are in a muddy field full of cabbages.
Alias +: FIELD
Exits: 0, 0, 3001, 3003, 3002, 0, 2001, 0, 0, 0

----- Some cabbages -> Narrative Object -----

Title: Some cabbages
Description: These are your average cabbage, if some what
on the small size. Available in a range of greens,
yellows and browns.
Alias +: CABBAGE, CABBAGES
Location: 3000

----- Non Player Barrier -----

Location Index: 3000
Direction Blocked: 16
Block Alias +: FARMER
Pass Alias +:

(3001) A cabbage field. -> Basic Location =====

Location Number: 3001
Title: A cabbage field.
Description: You are in a muddy field full of cabbages.
Alias +: FIELD
Exits: 0, 0, 0, 0, 3003, 3002, 3000, 0, 0, 0

----- Some cabbages -> Narrative Object -----

Title: Some cabbages
Description: These are your average cabbage, if some what
on the small size. Available in a range of greens,
yellows and browns.
Alias +: CABBAGE, CABBAGES
Location: 3001

(3002) A cabbage field. -> Basic Location =====

Location Number: 3002
Title: A cabbage field.
Description: You are in a muddy field full of cabbages.
Alias +: FIELD
Exits: 3000, 3001, 3003, 0, 0, 0, 2002, 0, 0, 0

----- Some cabbages -> Narrative Object -----

Title: Some cabbages
Description: These are your average cabbage, if some what
on the small size. Available in a range of greens,
yellows and browns.
Alias +: CABBAGE, CABBAGES
```

Appendix A. Text Dump of the Cabbage Zone

```
Location: 3002

----- Non Player Barrier -----

Location Index: 3002
Direction Blocked: 16
Block Alias +: FARMER
Pass Alias +:

(3003) A cabbage field. -> Basic Location =====

Location Number: 3003
Title: A cabbage field.
Description: You are in a muddy field full of cabbages.
Alias +: FIELD
Exits: 3001, 0, 0, 0, 0, 0, 3002, 3000, 0, 0

----- Some cabbages -> Narrative Object -----

Title: Some cabbages
Description: These are your average cabbage, if some what
on the small size. Available in a range of greens,
yellows and browns.
Alias +: CABBAGE, CABBAGES
Location: 3003

(3003) A Cabbage Farmer -> Creature =====

Name: A Cabbage Farmer
Description: This is one of the local town's people whos
job is to grow and tend the cabbages for the town.
Alias +: FARMER, MAN
Enter Message: A Farmer walks in.
Exit Message: A Farmer wolks off.
Actions +: The Farmer tends to the cabbages.,
The Farmer starts to hoe the ground.,
The Farmer looks around the field.,
The Farmer rests against his hoe.
Location: 3003
Reset Frequency: 60
Health: 30
Combat frequency: 0
Combat Actions +: {0} hit[/s] {3} with a ham sized fist.,
{0} catch[/es] {3} with a stunning blow.,
{0} lash[/es] out at {3} hitting hard.
Aggressiveness: 0
Money: 0
Gender: 0
Carry Encumbrance: 1000
Encumbrance: 1500
Friends +:
Enemies +:

----- (0) A Rough Shirt -> Worn Object -----

Title: A Rough Shirt
Description: This is a rather shabby, rough, long sleeved
cotton shirt.
Alias +: SHIRT, ROUGH
Reset Message:
Location: 0
Reset Frequency: 0
Value: 10
Reset on Pickup: false
Encumbrance: 10
Required Body Parts: 5, 6, 7, 7, 8, 8, 4

----- (0) Some Leather Trousers -> Worn Object -----

Title: Some Leather Trousers
Description: This is a pair of soft leather trousers, a
```

```
bit worn but serviceable.
Alias +: TROUSERS, LEATHER
Reset Message:
Location: 0
Reset Frequency: 0
Value: 15
Reset on Pickup: false
Encumbrance: 15
Required Body Parts: 11, 12, 12, 13, 13

----- (0) A Pair of Leather Boots -> Worn Object -----

Title: A Pair of Leather Boots
Description: This is a pair of leather boots, they have
seen better days and one has a hole in the bottom.
Better than bear feet though
Alias +: BOOTS, LEATHER, PAIR
Reset Message:
Location: 0
Reset Frequency: 0
Value: 15
Reset on Pickup: false
Encumbrance: 10
Required Body Parts: 15, 15, 14, 14

----- (0) A small Sack -> Container -----

Title: A small Sack
Description: This is a small cloth sack.
Alias +: SACK, CLOTH
Reset Message:
Location: 0
Reset Frequency: 0
Value: 5
Reset on Pickup: false
Encumbrance: 5
Maximum Capacity: 20

----- (0) Some Bread -> Food -----

Title: Some Bread
Description: This is a chunk of fresh bread. Just the
thing to fill an empty stomach.
Alias +: BREAD
Reset Message:
Location: 0
Reset Frequency: 0
Value: 10
Reset on Pickup: false
Encumbrance: 10
Hour Value: 12

----- (0) Some Cheese -> Food -----

Title: Some Cheese
Description: This is a piece of yellow cheese, except
for the odd spot of green that is.
Alias +: CHEESE
Reset Message:
Location: 0
Reset Frequency: 0
Value: 5
Reset on Pickup: false
Encumbrance: 5
Hour Value: 4

----- (0) A Hoe -> Wielded Weapon -----

Title: A Hoe
Description: This is a regular hoe commonly used by pesants
toiling in the fields. It is a slendar wooden pole
```



```
about five foot in length and has a small iron blade
at one end.
Alias +: HOE
Reset Message:
Location: 0
Reset Frequency: 0
Value: 40
Reset on Pickup: false
Encumbrance: 70
Required Body Parts: 10, 10
Fixed Damage: 2
Random Damage: 2
Combat Actions +: {0} swing[/s] a hoe at {3}.,
                  {0} lunge[/s] at {3} with a hoe hitting {1},
                  {0} jab[/s] at {3} with {4} hoe.

----- Body Parts -----

      Body Parts: 0, 1, 1, 2, 3, 4, 4, 5, 6, 7, 7, 8, 8, 9, 9,
10, 10, 11, 12, 12, 13, 13, 14, 14, 15, 15, 19, 19,
19, 19, 19, 19, 19, 19, 20, 20

----- Base Damage -----

      Natural Fixed Damage: 2

----- Base Damage Modifier -----

      Natural Random Damage: 2

----- Base Hit Chance -----

      Natural Hit Chance: 20
```

Figure A-1. Dump to Text for Cabbage Zone

Appendix B. License Agreement

Last Updated: 11th November 2002

WolfMUD is an original work:

Copyright © 1999-2003 Andrew "Diddymus" Rolfe, WolfMUD.org

email:

Diddymus@WolfMUD.org

Diddymus@CodeCrafterStudio.com

WolfMUD is licensed software. It is not Public Domain, Shareware or anything similar. This file contains the terms and conditions of the WolfMUD License Agreement. This Agreement grants you a non-exclusive license to use the software as outlined below. If you wish to use WolfMUD in any way, or use any of its source code, you must read this license and are legally bound to comply with it.

Use of the WolfMUD code in any capacity implies that you have read, understood, and agreed to abide by the terms and conditions set down by this license. If you use WolfMUD without complying with the license, it is a breach of the law.

You must not use WolfMUD to make money in any way or be otherwise compensated. WolfMUD was developed in people's uncompensated spare time and was given away free of charge, and you must not use it to make money. WolfMUD must not be used in any way to accept or demand fees, donations, or other compensation of any kind.

Examples include, but are not limited to the following:

- You must NOT require players to pay a fee or donation for using WolfMUD.
- You must NOT accept bribes in any form for levels, equipment or anything else that have not been aquired through hard play.
- You must NOT accept a fee or donation for the running of a WolfMUD server or for passing on the cost of service providers.
- You cannot sell, distribute or exchange WolfMUD for a fee or donation. It must be free to any at all times.

You must give credit to the authors of WolfMUD. It is a project created in spare time for no compensation and is given away at no charge. Therefore the only recognition the authors get is their name in the credits and source code. If this is removed and you claim to have written the code yourself it shows you have no respect for the time and effort involved in creating software.

The following MUST be followed if you wish to use or modify the WolfMUD codebase:

- This license must be distributed 'AS IS' with all copies or portions of WolfMUD that you distribute, if any, including works derived from WolfMUD.
- You MUST NOT remove any copyright notice or authors name from the source code.
- The text for the CREDITS command must remain intact and unmodified. The only exception is that you may add ADDITIONAL text if you make modifications.
- The initial text seen by players when they connect must remain intact and unmodified. The only exception is that you may add ADDITIONAL text if you make modifications.
- The text for the WolfMUD command must remain intact and unmodified. There are no exceptions to this requirement.

- WolfMUD has been written in 100% Pure Java(TM), making it available on a vast number of platforms. WolfMUD MUST stay 100% Pure Java(TM). Modifications are NOT allowed to tie WolfMUD to any one specific Platform or Operating System.¹
- If you make modification to the source code it should be clearly documented as to the modifications made and the date of the modification.

Permission is given to use, modify and redistribute all WolfMUD source code and documentation as long as such use does not violate any of the rules set down by this license. If you cannot abide by any part of this License Agreement then DO NOT USE WolfMUD.

If you are running a server based on WolfMUD, please contact the Authors at the top of this license agreement with some details.

If you write an article about WolfMUD, please contact the Authors at the top of this license agreement with some details and gain permission before publication.

If you find any bugs, wish to make comemnts etc. please contact the Authors at the top of this license agreement.

DISCLAIMER

THE CREATORS OF WOLFMUD MAKE NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. THE CREATORS OF WOLFMUD SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES.

-- Andrew 'Diddymus' Rolfe

Trademarks:

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and/or other countries.

Notes

1. Platform specific modifications such as scripts for administration and native code for performance improvements should be made available as an *OPTIONAL* companion distribution.

Colophon

This document was produced 100% on a Linux system running Kernel versions 2.4.18 and 2.4.19.

The source of this document was written using DocBook DSSSL version 1.76-1 and DocBook DTDs version 1.0-8 covering DocBook 4.1.2. The DocBook utils version used was 0.6.9-25 with jw version 1.1. The parser was OpenJade version 1.3.1-4.

Previews of the document were created in PostScript format using docbook2ps.

Final document production was done using docbook2dvi and running the DVI files through dvipdfm version 0.13.2c for the PDF document. For the text version docbook2html was used and the output run through Lynx version 2.8.4dev.16.

The document SGML source was edited using the VIM editor version 6.0z ALPHA.

Screenshots were taken using GIMP version 1.2.1. Diagrams were produced using Dia version 0.88.1.

The WolfMUD Builder was run using the Sun JDK version 1.3.1-b24 for the purposes of the screenshots. The window manager used was FVWM2 version 2.2.4.

The document was previewed using gv 3.5.8. Acrobat Reader 4.0 and 5.0.5 were used to preview final document versions.

Document and Image version control was done with CVS version 1.11.