The UOX Hackers' Guide

UOX Freelancers Group

Valuable information for beginners, a reference guide to developers.

Names of all products herein are used for identification purposes only and are trademarks and/or registered trademarks of their respective owners. The authors makes no claim of ownership or corporate association with the products or companies that own them.

Contents

Chapter 1

Introduction

1.1   About this Document

This document is intended to be used by beginners who want to understand and/or contribute to UOX project. Most of the protocol information in this guide was taken from Jerrith's UO Packets Guide (Thanks Jerrith)

1.2   What is UOX

UOX stands for **U**ltima **O**ffline e**X**perience. And it allows one to play Ultima Online over a LAN, Modem Connection or just locally creating your own custom world.

But, of course, UOX's most wonderfull caracteristic is it's source Open and avaliable to anyone willing to use and contribute.

1.3   How to contribute

There are several ways to contribute to UOX development and grow:

- Sending bug reports;

- Updating and improving default scripts;

- Updating and improving documentation;

- Creating and updating UOX tools;

- Coding the server itself;

The server source is in C/C++, but as you can see above, it's not needed to know how to use a programming language to contribute to the project.

1.4   How do I compile it? or Where to get a compiler?

UOX is intended to follow ANSI standard to keep the source portable to other Operation Systems and compilers. If you are looking for a free compiler, you can use Mingw, a free Win32 port of GNU's gcc compiler.

Chapter 2

UO Client/Server Protocol

2.1 Game Packets Structure

In every packet from or to client, the very first byte is a command identifier, represented here as cmd into packet's structure.

For simplicity, bytes are used, in this document. If the size of a piece of data is larger than a byte, the size will be written afterwards, enclosed in []'s. Sometimes, unknown or default values will be enclosed in ()'s.

2.1.1 Create Character [0x00]

This message is sent from UO client when user selects to create a new character. Since this packet comes with character's initial attributes, it's important to check for consistency before using information contained in there.

```
Create Character (104 bytes)
BYTE cmd
BYTE[4] pattern1 (0xededeced)
BYTE[4] pattern2 (0xffffffff)
BYTE pattern3 (0x00)
BYTE[30] char name
```

```
BYTE[30] char password
BYTE sex (0=male, 1=female)
BYTE str
BYTE dex
BYTE int
BYTE skill1
BYTE skill1value
BYTE skill2
BYTE skill2value
BYTE skill3
BYTE skill3value
BYTE[2] skinColor
BYTE[2] hairStyle
BYTE[2] hairColor
BYTE[2] facial hair
BYTE[2] facial hair color
BYTE[2] location // from starting list
BYTE[2] unknown1
BYTE[2] slot
BYTE[4] clientIP
BYTE[2] shirt color
BYTE[2] pants color
```

Notes: Str, dex and int should always sum to 65. Str, dex and int should always be between 10 and 45, inclusive. Skill1, skill2, and skill3 should never be the same value. Skill1, skill2, and skill3 should always be between 0 and 45, inclusive.[1] Skill1value, skill2value, and skill3value should always sum to 100. Skill1value, skill2value, and skill3value should always be between 0 and 50, inclusive. SkinColor should always be between 0x3EA and 0x422, exclusive. HairColor and facialHairColor should always be between 0x44E and 0x4AD, exclusive. HairStyle should be between 0x203B and 0x204A, exclusive, and it should also exclude 0x203D to 0x2044, exclusive. FacialHairStyle should be between 0x203E and 0x204D Shirt color and Pants color need bounds checking too.

---

[1]Check Skills table for the meaning skill numbers

2.1.2 Disconnect Notification [0x01]

This is a message sent from client to server when the user chooses to return to the main menu from the character select menu. Since the character select menu no longer has a main menu button, this message is no longer used.

```
Disconnect Notification (5 bytes)
BYTE cmd
BYTE[4] pattern (0xFFFFFFFF)
```

2.1.3 Move Request [0x02]

Client message sent when user is walking/running.

```
Move Request (7 bytes)
BYTE cmd
BYTE direction
BYTE sequence number
BYTE[4] fastwalk prevention key
```

Note: Sequence number starts at 0, is reseted when reaches 255. However, when it's reseted, the next sequence number is 1, not 0.

2.1.4 Talk Request [0x03]

```
Talk Request (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE SpeechType
BYTE[2] Color
BYTE[2] SpeechFont
BYTE[x] msg - Null Terminated String.
```

blockSize is calculated as length of msg (x) + 8.

2.1.5   Unknown Request [0x04]

```
Request (2 bytes)
BYTE cmd
BYTE mode (0=off, 1=on)
```

2.1.6   Attack Request [0x05]

Client message sent when user tries to attack someone.

```
Attack Request (5 bytes)
BYTE cmd
BYTE[4] ID to be attacked
```

2.1.7   Double click [0x06]

```
Double click (5 bytes)
BYTE cmd
BYTE[4] ID of double clicked object
```

2.1.8   Pick up Item(s) [0x07]

```
Pick Up Item(s) (7 bytes)
BYTE cmd
BYTE[4] item id
BYTE[2] # of items in stack
```

2.1.9   Drop Item(s) [0x08]

```
Drop Item(s) (14 bytes)
BYTE cmd
BYTE[4] item id
BYTE[2] X Location
BYTE[2] Y Location
BYTE Z Location
```

```
BYTE[4] Move Into Container ID (FF FF FF FF if normal world)
```

2.1.10   Single click [0x09]

```
Single click (5 bytes)
BYTE cmd
BYTE[4] ID of single clicked object
```

2.1.11   Stat window info [0x11]

Message sent by the server

```
Stat window info (66 bytes)
BYTE cmd
BYTE[2] packet length (0x0042)
BYTE[4] player id
BYTE[30] playerName
BYTE[2] currentHitpoints
BYTE[2] maxHitpoints
BYTE[1] name change flag (0xFF = allowed, 0 = not allowed)
BYTE[1] flag (0x00 - Stats after this flag are invalid.  0x01 - Stats after this flag are valid)
BYTE sex (0=male, 1=female)
BYTE[2] str
BYTE[2] dex
BYTE[2] int
BYTE[2] currentStamina
BYTE[2] maxStamina
BYTE[2] currentMana
BYTE[2] maxMana
BYTE[4] gold
BYTE[2] armor class
BYTE[2] weight
```

   Note:For characters other than the player, currentHitpoints and maxHitpoints
are not the actual values. MaxHitpointsis a fixed value, and currentHitpoints works
like a percentage.

2.1.12   Request Skill/Action/Magic Usage [0x12]

```
Request Skill/Action/Magic Usage (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE type
0x24 ($) - skill
BYTE[blockSize-4] skill (null terminated strings)
"1 0"- anatomy
"2 0"- animal lore
"3 0"- item identification
"4 0"- arms lore
"6 0"- begging
"9 0"- peacemaking
"12 0"- cartography
"14 0"- detect hidden
"15 0"- entice
"16 0"- evaluate intelligence
"19 0"- forensic evaluation
"21 0"- hiding
"22 0"- provocation
"23 0"- inscription
"30 0"- poisoning
"32 0"- spirit speak
"33 0"- stealing
"35 0"- taming
"36 0"- taste identification
"38 0"- tracking
0x56 (V) - Macro'd Spell
BYTE[blockSize-4] Spell (null terminated strings)
"2"- Create Food
"3"- Feeblemind
"4"- Heal
"5"- Magic Arrow
"6"- Night Sight
"7"- Reactive Armor
```

```
"8"- Weaken
"9"- Agility
"10"- Cunning
"11"- Cure
"12"- Harm
"13"- Magic Trap
"14"- Magic Untrap
"15"- Protection
"16"- Strength
"17"- Bless
"18"- Fireball
"19"- Magic Lock
"20"- Poison
"21"- Telekenisis
"22"- Teleport
"23"- Unlock
"24"- Wall of Stone
"25"- Arch Cure
"26"- Arch Protection
"27"- Curse
"28"- Fire Field
"29"- Greater Heal
"30"- Lightning
"31"- Mana Drain
"32"- Recall
"33"- Blade Spirit
"34"- Dispel Field
"35"- Incognito
"36"- Reflection
"37"- Mind Blast
"38"- Paralyze
"39"- Poison Field
"40"- Summon Creature
"41"- Dispel
"42"- Energy Bolt
"43"- Explosion
```

```
"44"- Invisibility
"45"- Mark
"46"- Mass Curse
"47"- Paralyze Field
"48"- Reveal
"49"- Chain Lightning
"50"- Energy Field
"51"- Flame Strike
"52"- Gate
"53"- Mana Vampire
"54"- Mass Dispel
"55"- Meteor Shower
"56"- Polymorph
"57"- Earthquake
"58"- Energy Vortex
"59"- Ressurection
"60"- Summon Air Elemental
"61"- Summon Daemon
"62"- Summon Earth Elemental
"63"- Summon Fire Elemental
"64"- Summon Water Elemental
0x58 (X) - Open Door
BYTE null termination (0x00)
0xc7 - action
BYTE[blockSize-4] Action (null terminated strings)
"bow"
"salute"
```

2.1.13   Drop - Wear Item [0x13]

```
Drop->Wear Item (10 bytes)
BYTE cmd
BYTE[4] itemid
BYTE layer (see layer list at top)
BYTE[4] playerID
```

Note: The layer byte should not be trusted.

2.1.14 Object Information [0x1A]

```
Object Information (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] itemID
BYTE[2] model #
if (itemID & 0x80000000)
BYTE[2] item count (or model # for corpses)
if (model & 0x8000)
BYTE Incr Counter (increment model by this #)
BYTE[2] xLoc (only use lowest significant 15 bits)
BYTE[2] yLoc
if (xLoc & 0x8000)
BYTE direction
BYTE zLoc
if (yLoc & 0x8000)
BYTE[2] dye
if (yLoc & 0x4000)
BYTE flag byte (See Apendix)
```

2.1.15 Char Location and body type [0x1B]

```
Char Location and body type (37 bytes)
BYTE cmd
BYTE[4] player id
BYTE[4] unknown1
BYTE[2] bodyType
BYTE[2] xLoc
BYTE[2] yLoc
BYTE[2] zLoc
BYTE direction
BYTE[2] unknown2
```

```
BYTE[4] unknown3 (usually has FF somewhere in it)
BYTE[4] unknown4
BYTE flag byte
BYTE highlight color
BYTE[7] unknown5
```

2.1.16   Send Speech [0x1C]

```
Send Speech (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] itemID (FF FF FF FF = system)
BYTE[2] model (item hex # - FF FF = system)
BYTE Type
BYTE[2] Text Color
BYTE[2] Font
BYTE[30] Name
BYTE[?] Null-Terminated Message (? = blockSize - 44)
```

2.1.17   Delete object [0x1D]

```
Delete object (5 bytes)
BYTE cmd
BYTE[4] item/char id
```

2.1.18   Draw Game Player [0x20]

```
Draw Game Player (19 bytes)
BYTE cmd
BYTE[4] creature id
BYTE[2] bodyType
BYTE unknown1 (0)
BYTE[2] skin color / hue
BYTE flag byte
BYTE[2] xLoc
```

```
BYTE[2] yLoc
BYTE[2] unknown2 (0)
BYTE direction
BYTE zLoc
```

Note: Only used with the character being played by the client.

2.1.19   Character Move Reject [0x21]

Message sent by server when a movement request is rejected.

```
Character Move Reject (8 bytes)
BYTE cmd
BYTE sequence #
BYTE[2] xLoc
BYTE[2] yLoc
BYTE direction
BYTE zLoc
```

2.1.20   Character Move ACK/ Resync Request [0x22]

```
Character Move ACK/ Resync Request(3 bytes)
BYTE cmd
BYTE sequence (matches sent sequence)
BYTE (0x00)
```

2.1.21   Dragging of Items [0x23]

```
Dragging of Items (26 bytes)
BYTE cmd
BYTE[2] model #
BYTE[3] unknown1
BYTE[2] stack count
BYTE[4] Source ID
BYTE[2] Source xLoc
```

```
BYTE[2] Source yLoc
BYTE Source zLoc
BYTE[4] Target id
BYTE[2] Target xLoc
BYTE[2] Target yLoc
BYTE Target zLoc
```

2.1.22   Draw Container [0x24]

```
Draw Container (7 bytes)
BYTE cmd
BYTE[4] item id
BYTE[2] model-Gump
0x003c = backpack
```

2.1.23   Add Item to Container [0x25]

```
Add Item to Container (20 bytes)
BYTE cmd
BYTE[4] item id to add
BYTE[2] model
BYTE unknown1 (0)
BYTE[2] # of items
BYTE[2] xLoc in container
BYTE[2] yLoc in container
BYTE[4] itemID of container
BYTE[2] color
```

2.1.24   Kick Player [0x26]

```
Kick Player (5 bytes)
BYTE cmd
BYTE[4] ID of GM who issued kick?
```

Note: Server Message

2.1.25   Reject Request to move Items [0x27]

```
Kick Player (5 bytes)
BYTE cmd
BYTE[4] ID of GM who issued kick?
```

Note: Server Message

2.1.26   Clear Square [0x28]

```
Clear Square (5 bytes)
BYTE cmd
BYTE[2] xLoc
BYTE[2] yLoc
```

Note: Server Message

2.1.27   Paperdoll Clothing Added [0x29]

```
Paperdoll Clothing Added(1 bytes)
BYTE cmd
```

Note: Server Message

2.1.28   Resurrection Menu Choice [0x2C]

Client and Server message. Note: Resurrection menu has been removed from UO, thus, this message is currently unused.

```
Resurrection Menu Choice (2 bytes)
BYTE cmd
BYTE action (2=ghost, 1=resurrect, 0=from server)
Note: Resurrection menu has been removed from UO.
```

2.1.29   Worn Item [0x2E]

```
Worn Item (15 bytes)
BYTE cmd
BYTE[4] itemid (always starts 0x40 in my data)
BYTE[2] model (item hex #)
BYTE (0x00)
BYTE layer
BYTE[4] playerID
BYTE[2] color/hue
```

2.1.30   Fight Occurring [0x2F]

This packet is sent when there is a fight going on somewhere on screen.

```
Fight Occurring (10 bytes)
BYTE cmd
BYTE unknown1 (0)
BYTE[4] ID of attacker
BYTE[4] ID of attacked
```

2.1.31   Pause/Resume Client [0x33]

```
Pause/Resume Client (2 bytes)
BYTE cmd
BYTE pause/resume (0=pause, 1=resume)
```

2.1.32   Get Player Status [0x34]

```
Get Player Status (10 bytes)
BYTE cmd
BYTE[4] pattern (0xededeeded)
BYTE getType
0x04 - Basic Stats (Packet 0x11 Response)
0x05 = Request Skills (Packet 0x3A Response)
```

```
BYTE[4] playerID
```

2.1.33   Send Skills/Lock state [0x3A]

```
Server Version - Send Skills (Variable)
BYTE cmd
BYTE[2] blockSize
BYTE Type (0x00= full list, 0xFF = single skill update)
Repeat next until done - 46 skills
BYTE[2] id # of skill (0x01 - 0x2e)
BYTE[2] skill Value * 10
BYTE[2] Unmodified Value * 10
BYTE skillLock (0=up, 1=down, 2=locked)
BYTE[2] null (00 00)  (ONLY IF TYPE == 0x00)
```

   Note: Can also send just one skill, to update that skill.

```
Client Version - Set Skill Lock (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[2] id# of skill
BYTE skillLock (0=up, 1=down, 2=locked)
```

2.1.34   Buy Item(s) [0x3B]

```
Buy Item(s) (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] vendorID
BYTE flag
0x00 - no items following
0x02 - items following
For each item
BYTE (0x1A)
BYTE[4] itemID (from 3C packet)
BYTE[2] # bought
```

2.1.35   Items in Container [0x3C]

```
Items in Container (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[2] # of Item segments
Item Segments:
BYTE[4] itemID
BYTE[2] model
BYTE unknown1 (0x00)
BYTE[2] # of items in stack
BYTE[2] xLoc
BYTE[2] yLoc
BYTE[4] Container ItemID
BYTE[2] color
```

2.1.36   Personal Light Level [0x4E]

```
Personal Light Level (6 bytes)
BYTE cmd
BYTE[4] creature id
BYTE level
```

2.1.37   Overall Light Level [0x4F]

```
Overall Light Level (2 bytes)
BYTE cmd
BYTE level
0x00 - day
0x09 - OSI night
0x1F - Black
Max normal val = 0x1F
```

2.1.38   Idle Warning [0x53]

```
Idle Warning(2 bytes)
BYTE cmd
BYTE value (0x07 - idle, 0x05 - another character is online)
"Another character from this account is currently online in this world.  You must either log in as t
```

2.1.39   Play Sound Effect [0x57]

```
Play Sound Effect (12 bytes)
BYTE cmd
BYTE mode (0x00=quiet, repeating, 0x01=single normally played sound effect)
BYTE[2] SoundModel
BYTE[2] unknown3 (speed/volume modifier? Line of sight stuff?)
BYTE[2] xLoc
BYTE[2] yLoc
BYTE[2] zLoc
```

2.1.40   Login Complete, start game [0x55]

```
Login Complete, Start Game (1 byte)
BYTE cmd
```

2.1.41   Time [0x5B]

```
Time (4 bytes)
BYTE cmd
BYTE hour
BYTE minute
BYTE second
\end{veratim}

\subsection{Login Character [0x5D]}
\begin{verbatim}
Login Character (73 bytes)
```

```
BYTE cmd
BYTE[4] pattern1 (0xedededed)
BYTE[30] char name
BYTE[30] char password
BYTE[4] slot choosen (0-based)
BYTE[4] clientIP
```

Note: ClientIP should not be trusted.

2.1.42   Set Weather [0x65]

```
Set Weather (4 bytes)
BYTE cmd
BYTE type
        0x00 - "It starts to rain"
        0x01 - "A fierce storm approaches."
        0x02 - "It begins to snow"
        0x03 - "A storm is brewing.",
        0xFF - None (turns off sound effects),
        0xFE (no effect?? Set temperature?)
BYTE num (number of weather effects on screen)
BYTE temperature
```

Note: Temperature has no effect at present. Note: maximum number of weather effects on screen is 70. Note: If it is raining, you can add snow by setting the num to the num of rain currently going, plus the number of snow you want. Note: Weather messages are only displayed when weather starts. Note: Weather will end automatically after 6 minutes without any weather change packets. Note: You can totally end weather (to display a new message) by teleporting. I think it's either the 0x78 or 0x20 messages that reset it, though I haven't checked to be sure (other possibilities, 0x4F or 0x4E)

2.1.43   Book - Page [0x66]

```
Books - Page (Variable # of bytes)
```

```
BYTE cmd
BYTE[2] blockSize
BYTE[4] bookID
BYTE[2] # of pages in this packet (when requesting always requests 1)
For each page:
        BYTE[2] page number
        BYTE[2] number of lines on page
                Server -> (-1 = no lines)
                Client -> (-1 = request page)   (No longer used)
                Client -> (> 0 = write page)
        Repeated for each line:
                BYTE[var] null terminated line
```

Note: Number of lines on page is always -1 for page requests

2.1.44 Targeting Cursor Commands [0x6C]

```
Targeting Cursor Commands (19 bytes)
BYTE cmd
BYTE type
        0x00 = Select Object
        0x01 = Select X, Y, Z
BYTE[4] cursorID
BYTE Cursor Type
        Always 0 now
The following are always sent but are only valid if sent by client
BYTE[4] Clicked On ID
BYTE[2] click xLoc
BYTE[2] click yLoc
BYTE unknown (0x00)
BYTE click zLoc
BYTE[2] model # (if a static tile, 0 if a map/landscape tile)
```

Note: the model number shouldn't be trusted.

2.1.45   Play Midi Music [0x6D]

```
Play Midi Music (3 bytes)
BYTE cmd
BYTE[2] musicID
```

2.1.46   Character Animation [0x6E]

```
Character Animation (14 bytes)
BYTE cmd
BYTE[4] item/char ID
BYTE[2] movement model
        0x00 = walk
        0x01 = walk faster
        0x02 = run
        0x03 = run (faster?)
        0x04 = nothing
        0x05 = shift shoulders
        0x06 = hands on hips
        0x07 = attack stance (short)
        0x08 = attack stance (longer)
        0x09 = swing (attack with knife)
        0x0a = stab (underhanded)
        0x0b = swing (attack overhand with sword)
        0x0c = swing (attack with sword over and side)
        0x0d = swing (attack with sword side)
        0x0e = stab with point of sword
        0x0f = ready stance
        0x10 = magic (butter churn!)
        0x11 = hands over head (balerina)
        0x12 = bow shot
        0x13 = crossbow
        0x14 = get hit
        0x15 = fall down and die (backwards)
        0x16 = fall down and die (forwards)
        0x17 = ride horse (long)
```

```
        0x18 = ride horse (medium)
        0x19 = ride horse (short)
        0x1a = swing sword from horse
        0x1b = normal bow shot on horse
        0x1c = crossbow shot
        0x1d = block #2 on horse with shield
        0x1e = block on ground with shield
        0x1f = swing, and get hit in middle
        0x20 = bow (deep)
        0x21 = salute
        0x22 = scratch head
        0x23 = 1 foot forward for 2 secs
        0x24 = same
BYTE unknown1 (0x00)
BYTE direction
BYTE[2] repeat (1 = once / 2 = twice / 0 = repeat forever)
BYTE forward/backwards(0x00=forward, 0x01=backwards)
BYTE repeat Flag (0 - Don't repeat / 1 repeat)
BYTE frame Delay (0x00 - fastest / 0xFF - Too slow to watch)
```

2.1.47 Secure Trading [0x6F]

```
Secure Trading (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE action
BYTE[4] id1
BYTE[4] id2
BYTE[4] id3
BYTE nameFollowing (0 or 1)
If (nameFollowing = 1)
BYTE[?] charName
```

2.1.48   Graphical Effect [0x70]

```
Graphical Effect (28 bytes)
BYTE cmd
BYTE direction type
        00 = go from source to dest
        01 = lightning strike at source
        02 = stay at current x,y,z
        03 = stay with current source character id
BYTE[4] character id
BYTE[4] target id
BYTE[2] model of the first frame of the effect
BYTE[2] xLoc
BYTE[2] yLoc
BYTE zLoc
BYTE[2] xLoc of target
BYTE[2] yLoc of target
BYTE zLoc of target
BYTE speed of the animation
BYTE duration (0=really long, 1= shortest)
BYTE[2] unknown2 (0 works)
BYTE adjust direction during animation (1=no)
BYTE explode on impact
```

2.1.49   Bulletin Board Message [0x71]

```
Bulletin Board Message (Variable # of bytes)
BYTE cmd
BYTE[2] len
BYTE subcmd
BYTE[len-4] submessage
```

Submessage 0 - Display Bulletin Board

Server Message.

```
Submessage 0 - Display Bulletin Board
BYTE[4] BoardID
BYTE[22] board name (default is "bulletin board", the rest nulls)
BYTE[4] ID (0x402000FF)
BYTE[4] zero (0)
```

Submessage 1 - Message Summary

Server Message.

```
Submessage 1 - Message Summary
BYTE[4] BoardID
BYTE[4] MessageID
BYTE[4] ParentID (0 if top level)
BYTE posterLen
BYTE[posterLen] poster (null terminated string)
BYTE subjectLen
BYTE[subjectLen] subject (null terminated string)
BYTE timeLen
BYTE[timeLen] time (null terminated string with time of posting) ("Day 1 @ 11:28")
```

Submessage 2 - Message

```
Submessage 2 - Message
BYTE[4] BoardID
BYTE[4] MessageID
BYTE posterLen
BYTE[posterLen] poster (null terminated string)
BYTE subjectLen
BYTE[subjectLen] subject (null terminated string)
BYTE timeLen
BYTE[timeLen] time (null terminated string with time of posting) ("Day 1 @ 11:28")
BYTE[29] constant: (01 91 84 0A 06 1E FD 01 0B 15 2E 01 0B 17 0B 01 BB 20 46 04 66 13 F8 00 00 0E 75
BYTE numlines
For each line:
        BYTE linelen
```

```
        BYTE[linelen] body (null terminated)
```

Submessage 3 - Request Message

Client Message.

```
Submessage 3 - Request Message
BYTE[4] BoardID
BYTE[4] MessageID
```

Submessage 4 - Request Message Summary

Client Message.

```
Submessage 4 - Request Message Summary
BYTE[4] BoardID
BYTE[4] MessageID
```

Submessage 5 - Post a message

Client Message.

```
BYTE[4] BoardID
BYTE[4] Replying to ID (0 if this is a top level / non-reply post)
BYTE subjectLen  (length of the subject, includes null termination)
BYTE[subjectLen] subject (null terminated)
BYTE numlines
For each line:
        BYTE linelen
        BYTE[linelen] body (null terminated)
```

Submessage 6 - Remove Posted Message

Client Message.

```
BYTE[4] BoardID
BYTE[4] MessageID
```

2.1.50   Request War Mode Change/Send War Mode status [0x72]

Client/Server message, server replies with 0x77 packet

```
Request War Mode Change/Send War Mode status (5 bytes)
BYTE cmd
BYTE flag
        0x00 - Normal
        0x01 - Fighting
BYTE[3] unknown1 (always 00 32 00 in testing)
```

2.1.51   Ping message [0x73]

```
Ping message (2 bytes)
BYTE cmd
BYTE seq
```

2.1.52   Open Buy Window [0x74]

```
Open Buy Window (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] (vendorID | 0x40000000)
BYTE # of items
# of items worth of item segments
        BYTE[4] price
        BYTE length of text description
        BYTE[text length] item description
```

NOTE: This packet is always preceded by a describe contents packet (0x3c) with the container id as the (vendorID — 0x40000000) and then an open container packet (0x24?) with the vendorID only and a model number of 0x0030 (probably the model number for the buy screen)

2.1.53   Rename Character [0x75]

```
Rename Character (35 bytes)
BYTE cmd
BYTE[4] id
BYTE[30] new name
```

2.1.54   New Subserver [0x76]

```
New Subserver (16 bytes)
BYTE cmd
BYTE[2] xLoc
BYTE[2] yLoc
BYTE[2] zLoc
BYTE[9] unknown
```

2.1.55   Update Player [0x77]

```
Update Player (17 bytes)
BYTE cmd
BYTE[4] player id
BYTE[2] model
BYTE[2] xLoc
BYTE[2] yLoc
BYTE zLoc
BYTE direction
BYTE[2] hue/skin color
BYTE flag (bit field)
BYTE highlight color
```

2.1.56   Draw Object [0x78]

```
Draw object (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
```

```
BYTE[4] itemID/playerID
BYTE[2] model (item hex #)
if (itemID & 0x80000000)
        BYTE[2] amount/Corpse Model Num
BYTE[2] xLoc (only 15 lsb)
BYTE[2] yLoc
if (xLoc & 0x8000)
        BYTE direction
BYTE zLoc
BYTE direction
BYTE[2] dye/skin color
BYTE flag
BYTE notoriety (2's complement signed)
if (BYTE[4] == 0x00 0x00 0x00 0x00)
        DONE
else loop this until above if statement is satisified
        BYTE[4] itemID
        BYTE[2] model (item hex # - only 15 lsb)
        BYTE layer
        if (model & 0x8000)
                BYTE[2] hue


\subsection{Open Dialog Box [0x7C]}
\begin{verbatim}
Open Dialog Box (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] dialogID (echo'd back to the server in 7d)
BYTE[2] menuid (echo'd back to server in 7d)
BYTE length of question
BYTE[length of question] question text
BYTE # of responses
Then for each response:
        BYTE[2] model id # of shown item (if grey menu -- then always 0x00 as msb)
        BYTE[2] unknown2 (00 00 check or not?)
        BYTE response text length
```

```
      BYTE[response text length] response text
```

2.1.57   Client response to Dialog [0x7D]

```
Client Response To Dialog (13 bytes)
BYTE cmd
BYTE[4] dialogID (echoed back from 7C packet)
BYTE[2] menuid (echoed back from 7C packet)
BYTE[2] 1-based index of choice
BYTE[2] model # of choice
BYTE[2] unknown1 (00 00)
```

2.1.58   Login Request [0x80]

```
Login Request (62 bytes)
BYTE cmd
BYTE[30] userid
BYTE[30] password
BYTE unknown1 (not usually 0x00 - so not NULL)
```

2.1.59   Login Denied [0x82]

```
Login Denied (2 bytes)
BYTE cmd
BYTE why
        0x00 = unknown user
        0x01 = account already in use
        0x02 = account disabled
        0x03 = password bad
        0x04 and higher = communications failed
```

2.1.60   Delete Character [0x83]

```
Delete Character (39 bytes)
BYTE cmd
```

```
BYTE[30] password
BYTE[4] charIndex
BYTE[4] clientIP
```

2.1.61   Resend Characters After Delete [0x86]

```
Resend Characters After Delete (304 bytes)
BYTE cmd
BYTE[2] blockSize
BYTE # of characters
Following repeated 5 times
        BYTE[30] character name
        BYTE[30] character password
```

2.1.62   Open Paperdoll [0x88]

```
Open Paperdoll (66 bytes)
BYTE cmd
BYTE[4] charid
BYTE[60] text
BYTE flag byte
```

2.1.63   Corpse Clothing [0x89]

```
Corpse Clothing (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] corpseID
        BYTE itemLayer
        BYTE[4] itemID
BYTE terminator (0x00)
Followed by a 0x3C message with the contents.
```

2.1.64   Connect to Game Server [0x8C]

```
Connect to Game Server (11 bytes)
BYTE cmd
BYTE[4] gameServer IP
BYTE[2] gameServer port
BYTE[4] new key
```

2.1.65   Map message [0x90]

```
Map message(19 bytes)
BYTE cmd
BYTE[4] key used
BYTE[2] gump art id (0x139D)
BYTE[2] upper left x location
BYTE[2] upper left y location
BYTE[2] lower right x location
BYTE[2] lower right y location
BYTE[2] gump width in pixels
BYTE[2] gump height in pixels
```

2.1.66   Game Server Login [0x91]

```
Game Server Login (65 bytes)
BYTE cmd
BYTE[4] key used
BYTE[30] sid
BYTE[30] password
```

2.1.67   Book - Title Page [0x93]

```
Books - Title Page (99 bytes)
BYTE cmd
BYTE[4] bookID
BYTE write flag
```

```
        0x00 - non-writable
        0x01 - writable
BYTE new flag
BYTE[2] # of pages
BYTE[60] title
BYTE[30] author
```

2.1.68   Dye Window [0x95]

```
Dye Window (9 bytes)
BYTE cmd
BYTE[4] itemID of dye target
BYTE[2] ignored on send, model on return
BYTE[2] model on send, color on return  (default on server send is 0x0FAB)
```

NOTE: This packet is sent by both the server and client.

2.1.69   House/Boat Placement [0x99]

Bring up house/board placement View.

```
Bring Up House/Boat Placement View (26 bytes)
BYTE cmd
BYTE request (0x01 from server, 0x00 from client)
BYTE[4] ID of deed
BYTE[12] unknown (all 0)
BYTE[2] multi model (item model - 0x4000)
BYTE[6] unknown (all 0)
```

2.1.70   Console Entry Prompt [0x9A]

```
Console Entry Prompt(16 bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] objectID
```

```
BYTE[4] prompt#
BYTE[4] 0=request/esc, 1=reply
BYTE[?] textstring (optional)
BYTE terminator (0x00)
```

2.1.71   Request Help [0x9B]

```
Request Help (258 bytes)
BYTE cmd
BYTE[257] (0x00)
```

2.1.72   Sell List [0x9E]

```
Sell List (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] shopkeeperID
BYTE[2] numItems
For each item, a structure containing:
        BYTE[4] itemID
        BYTE[2] itemModel
        BYTE[2] itemHue/Color
        BYTE[2] itemAmount
        BYTE[2] value
        BYTE[2] nameLength
        BYTE[?] name
```

2.1.73   Sell Reply [0x9F]

```
Sell Reply (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] shopkeeperID
BYTE[2] itemCount
For each item, a structure containing:
```

```
BYTE[4] itemID
BYTE[2] quantity
```

2.1.74   Select Server[0xA0]

```
Select Server (3 bytes)
BYTE cmd
BYTE[2] server # chosen
```

2.1.75   Update Current Health [0xA1]

```
Update Current Health (9 bytes)
BYTE cmd
BYTE[4] playerID
BYTE[2] maxHealth
BYTE[2] currentHealth
```

2.1.76   Update Current Mana [0xA2]

```
Update Current Mana (9 bytes)
BYTE cmd
BYTE[4] playerID
BYTE[2] maxMana
BYTE[2] currentMana
```

2.1.77   Update Current Stamina [0xA3]

```
Update Current Stamina (9 bytes)
BYTE cmd
BYTE[4] playerID
BYTE[2] maxStamina
BYTE[2] currentStamina
```

2.1.78 Client Machine info [0xA4]

```
Spy on Client (149 bytes)
BYTE cmd
BYTE[148] Unknown (previously, this has had info such as your graphics card name,
free HD space, number of processors, etc.)
```

2.1.79 Open Web Browser [0xA5]

```
Open Web Browser (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[blockSize-3] null terminated full web address
```

2.1.80 Tip/Notice window [0xA6]

```
Tips/Notice window (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE flag
        0x00 - tips window
        0x01 - notice window
BYTE[2] unknown1
BYTE[2] tip #
BYTE[2] msgSize
BYTE[?] message (? = blockSize - 10)

*Null terminated I think (Gimli)
```

2.1.81 Request Tips/Notice [0xA7]

```
Request Tips/Notice (4 bytes)
BYTE cmd
BYTE[2] last tip #
BYTE flag
```

```
        0x00 - tips window
        0x01 - notice window
```

2.1.82   Game Server List

```
Game Server List (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE System Info Flag
        0xCC - Don't send
        0x64 - Send Video card
        ?? -
BYTE[2] # of servers
Then each server --
        BYTE[2] serverIndex (0-based)
        BYTE[32] serverName
        BYTE percentFull
        BYTE timezone
        BYTE[4] pingIP
```

2.1.83   Characters/Starting Locations [0xA9]

```
Characters / Starting Locations (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE # of characters
Following repeated 5 times
        BYTE[30] character name
        BYTE[30] character password
        BYTE number of starting locations
Following for as many locations as you have
        BYTE locationIndex (0-based)
        BYTE[31] town (general name)
        BYTE[31] exact name
```

2.1.84   Attack Request Reply [0xAA]

```
OK / Not OK To Attack (5 bytes)
BYTE cmd
BYTE[4] CharID being attacked
```

```
ID is set to 00 00 00 00 when attack is refused.
```

2.1.85   Gump Text Entry Dialog [0xAB]

```
Gump Text Entry Dialog (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] id
BYTE parentID
BYTE buttonID
BYTE textlen
BYTE[?] text
BYTE cancel (0=disable, 1=enable)
BYTE style (0=disable, 1=normal, 2=numerical)
BYTE[4] format (if style 1, max text len, if style2, max numeric value)
BYTE text2len
BYTE[?] text2
```

2.1.86   Gump Text Entry Dialog Reply [0xAC]

```
Gump Text Entry Dialog Reply (Variable # of bytes)
BYTE cmd
BYTE[2] length
BYTE[4] ID
BYTE type
BYTE index
BYTE[3] unk
BYTE[?] reply
```

2.1.87 Unicode speech request [0xAD]

Note: This has been changed recently, the format shown here is the old one. It will be updated when I get the new format packet details

```
Unicode speech request (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE Type
BYTE[2] Color
BYTE[2] Font
BYTE[4] Language (Null Terminated)
        "enu"- United States English
BYTE[?][2] Msg - Null Terminated (blockSize - 12)
```

2.1.88 Unicode Speech message [0xAE]

```
Unicode Speech message(Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] ID
BYTE[2] Model
BYTE Type
BYTE[2] Color
BYTE[2] Font
BYTE[4] Language
BYTE[30] Name
BYTE[?][2] Msg - Null Terminated (blockSize - 48)
```

2.1.89 Display Death Action [0xAF]

```
Display Death Action (13 bytes)
BYTE cmd
BYTE[4] player id
BYTE[4] corpse id
BYTE[4] unknown (all 0)
```

2.1.90   Send Gump Menu Dialog

```
Send Gump Menu Dialog (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] id
BYTE[4] gumpid
BYTE[4] x
BYTE[4] y
BYTE[2] command section length
BYTE[?] commands (zero terminated)
BYTE[2] numTextLines
        BYTE[2] text length (in unicode (2 byte) characters.)
        BYTE[?] text (in unicode)
```

2.1.91   Gump Menu Selection [0xB1]

```
Gump Menu Selection (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] id
BYTE[4] gumpid
BYTE[4] choiceid
```

2.1.92   Chat Message [0xB2]

```
Chat Message (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[2] messageType
If message type==0x03EB (Display Enter Username window)
BYTE[8] unknown (all 0's)
If message type==0x03ED (Username accepted, display window)
BYTE[4] unknown (all 0's)
BYTE[?][2] unicode username
BYTE[4] unknown (all 0's)
```

```
If message type==0x03E8
BYTE[4] unknown (all 0's)
BYTE[?][2] unicode channel name
BYTE[2] unknown (0x0000)
BYTE[2] unknown (0x0030)
BYTE[2] unknown (0x0000)
```

2.1.93   Open Chat Window [0xB5]

This message is very incomplete. From the server, just know that it is 0xB5 len len, and pass the data through as is appropriate.

```
Open Chat window (64 bytes)
BYTE cmd
BYTE[63] chatname, if known by client (all 00 if unknown) (name in unicode)
```

2.1.94   Send Help/Tip Request [0xB6]

```
Send Help/Tip Request (9 bytes)
BYTE cmd
BYTE[4] id
BYTE[1] language # (1 for enu)
BYTE[3] language name (enu for English - United states)
```

2.1.95   Help/Tip Data [0xB7]

```
Help/Tip Data (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE[4] id
BYTE[?][2] message, in unicode
BYTE[2] null terminator (0x0000)
BYTE[2] message terminator (0x3300)
```

2.1.96   Request Char Profile [0xB8]

```
Request Char Profile (Variable # of bytes)
BYTE cmd
BYTE[2] blockSize
BYTE mode (CLIENT ONLY! Does not exist in server message.)
BYTE[4] id
If request, ends here.
If Update request
        BYTE[2] cmdType (0x0001 - Update)
        BYTE[2] msglen (# of unicode characters)
        BYTE[msglen][2] new profile, in unicode, not null terminated.
Else If from server
        BYTE[?] character name (not unicode, null terminated.)
        BYTE[2] (0x0000) (a non-unicode title string?)
        BYTE[?][2] profile (in unicode, ? can be 0)
        BYTE[2] (0x0000)
        BYTE[2] terminator (0x3300)
```

2.1.97   Enable Chat Button [0xB9]

Server Message.

```
Enable Chat Button (3 bytes)
BYTE cmd
BYTE unknown (0x00)
BYTE unknown2 (0x01)
```

2.1.98   Quest Arrow [0xBA]

Server Message.

```
Quest Arrow (6 bytes)
BYTE cmd
BYTE active (1=on, 0=off)
BYTE[2] xLoc
```

```
BYTE[2] yLoc
```

2.1.99   Ultima Messenger [0xBB]

Both client and server message.

```
Ultima Messenger  (9 bytes)
BYTE cmd
BYTE[4] id1
BYTE[4] id2
```

2.1.100   Seasonal Information [0xBC]

Server Message. If id2 == 1, then this is a season change. If it's a season change the id1 means: 0 - spring 1 - summer 2 - fall 3 - winter 4 - desolation

```
Seasonal Information(3 bytes)
BYTE cmd
BYTE id1
BYTE id2
```

2.1.101   Client Version Message [0xBD]

Client Message.

```
Client Version Message (Variable # of bytes)
BYTE cmd
BYTE[2] len
BYTE[len-3] string stating the client version (0 terminated)  (like: "1.26.4")
```

2.1.102   Misc. Commands Packet [0xBF]

This packet is send by both client and server, and starts with a command byte (0xBF), followed by two bytes for the length. After that, there is a two byte value which is a subcommand, and the message varies after that.

```
General Info (5 bytes, plus specific message)
BYTE cmd
BYTE[2] len
BYTE[2] subcmd
BYTE[len-5] submessage
```

Subcommand 1 - Initialize Fast Walk Prevention

Server Message. This sets up stack on the client and whenever it moves, it takes the
top value from this stack and uses it. (key1 start at the top, key6 at the botton).

```
Subcommand 1: Initialize Fast Walk Prevention [24 bytes (for 29 total)]
BYTE[4] key1
BYTE[4] key2
BYTE[4] key3
BYTE[4] key4
BYTE[4] key5
BYTE[4] key6
```

Subcommand 2 - Add key to Fast Walk Stack

Server message. This key is added tothe tp of the stack. In other words, it's the
one that will be used next. Basically, the other 5 only get used when the client is
sending moves faster than the server is responding.

```
Subcommand 2: Add key to Fast Walk Stack (4 bytes (for 9 total))
BYTE[4] newkey
```

Subcommand 5 - Unknown

Client Message.

```
Subcommand 5: Unknown (8 bytes (for 13 total))
BYTE[4] unknown (00 00 03 20)
BYTE[4] unknown (00 00 00 05)
```

Subcommand 6 - Party System

Party system is implemented as subsubcommands.

```
Subcommand 6: Party System (YES! Subsubcommands)
BYTE subsubcommand #
```

Subsubcommand 1 - Add a party member

Client Message.

```
Subsubcommand 1: Add a party member (4 bytes)
BYTE[4] id (if 0, a targeting cursor appears)
```

Subsubcommand 1 - Add party member(s)

Client Message.

```
Subsubcommand 1: Add party member(s) (1+ numMembers*4)
BYTE numMembers (total number of members in the party)
Then, for each member in numMembers:
        BYTE[4] id
```

Subsubcommand 2 - Remove a party member

When it's Client Message:

```
Subsubcommand 2: Remove a party member (4 bytes)
BYTE[4] id (if 0, a targeting cursor appears)
```

When it's a Server Message:

```
Subsubcommand 2: Remove a party member (? Bytes)
BYTE numMembers (total number of members in the new party)
BYTE[4] idofPlayerRemoved
Then, for each member in numMembers:
BYTE[4] id
```

Subsubcommand 3 - Send a message to party member

Client and Server message.

```
Subsubcommand 3: Tell party member a message (Variable # of bytes)
BYTE[4] id (of target, from client, of source, from server)
BYTE[n][2] Null terminated Unicode message.
```

Subsubcommand 4 - Send a message to all party members

Client message:

```
Subsubcommand 4: Tell full party a message (Variable # of bytes)
BYTE[n][2] Null terminated Unicode message.
```

    Server message:

```
Subsubcommand 4: Tell full party a message (Variable # of bytes)
BYTE[4] id (of source)
BYTE[n][2] Null terminated Unicode message.
```

Subsubcommand 6 - Party can loot me

Client message.

```
Subsubcommand 6: Party Can Loot Me?  (1 byte)
BYTE canloot (0=no, 1=yes)
```

Subsubcommand 8 - Accept join party invitation

Client message

```
Subsubcommand 8: Accept join party invitation (4 bytes)
BYTE[4] id (party leader's id)
```

Subsubcommand 9 - Decline join party invitation

Client message

```
Subsubcommand 9: Decline join party invitation (4 bytes)
BYTE[4] id (party leader's id)
```

Subcommand 8 - Set cursor hue

Server message.

```
Subcommand 8: Set cursor hue (1 byte (for 6 total))
BYTE hue  (0 = Felucca, unhued.  1 = Trammel, hued gold)
```

Subcommand 11 - Client language

Client message.

```
Subcommand 11: Client Language (3 bytes (for 8 total))
BYTE[3] language  (ENU, for English)
```

Subcommand 12 - Closed Status Gump

Server Message

```
Subcommand 12: Closed Status Gump (4 bytes (for 8 total))
BYTE[4] id (character id)
```

2.1.103   Predefined Message [0xC1]

Server Message

```
Predefined Message (Variable # of bytes (always 0x32 at present))
BYTE cmd
BYTE[2] len (0x32)
```

```
BYTE[4] id
BYTE[2] body
BYTE type (6 - lower left, 7 on player)
BYTE[2] hue
BYTE[2] font
BYTE[2] type (0x0007 so far)
BYTE[2] message number (0xA120 base)
        0xA12D - "You cannot use skills."
        0xA2E2 - "But that's not dead!"
        0xA5F3 - "Help request aborted."
        0xA5F0 - "Please enter a brief description (up to 40 characters) of
                  your problem:"
BYTE[0x20] - speaker's name
```

2.2   Patch Server Protocol

Connectiong to the patch server and issuing requests mst follow a specific order. First it connectes to compassion.owo.com:8888 (this can be changed in vercfg file). Then send a HelloRequest. The server will respond with a PatchServerTransfer. Disconnect from compassion and connectto the IP address and port returned to you. You will then send a HelloRequest to the new server, it will return a NoticeBlock.

   Now once you've connectd to the patch server and gotten the notice, you may start looking for patches. First, send a PatchListRequest for the first one, read in the PatchListData, then send the request for the second verson, and read in that data.

   Once you have a list of avaliable patches, you can request the actual patch by sending a PatchDataRequest. You will then receie the PatchData. When you have finished downloading the avaliable patches, you can apply them, and resend the PatchListReqiestuntil there aren't any more patches avaliable.

2.2.1   HelloRequest

```
Hello Request (4 bytes)
BYTE 0x15
```

```
BYTE 0x00
BYTE 0x00
BYTE 0x00
```

2.2.2   PatchServerTransfer

```
Patch Server Transfer (16 bytes)
BYTE 0x01
BYTE[7] (all bytes 0x00)
BYTE[4] IP Address
BYTE[4] Port
```

2.2.3   NoticeBlock

```
Notice Block (variable length)
BYTE 0x01
BYTE[3] (all bytes 0x00)
BYTE 0x01
BYTE[3] (all bytes 0x00)
BYTE 0x02
BYTE[3] (all bytes 0x00)
BYTE[4] Text Length
BYTE[textlength] Notice Data (NULL Terminated)
```

2.2.4   PatchListRequest

```
Patch List Request (variable length)
BYTE 0x01
BYTE[3] (all bytes 0x00)
BYTE[4] Version Text Lenght
BYTE[textlength] VersionText (ie. Win32_UO_Gold)
BYTE[4] Version
```

2.2.5   PatchListData

```
Patch List Data (variable length)
Repeat until done:
        BYTE[4] File name Length (0 = done)
        BYTE[filenamelength] Filename
        BYTE[4] File size
```

2.2.6   PatchDataRequest

```
Patch Data Request (variable length)
BYTE 0x02
BYTE[3] (all bytes 0x00)
BYTE[4] Filename length
BYTE[filenamelength] Filename
BYTE[4] (all bytes 0x00)
```

2.2.7   PatchData

```
BYTE[4] File name length
BYTE[filenamelength] File name
BYTE[4] File size
Repeat until done:
        BYTE[4] Block Length (0 = done)
        BYTE[blocklength] Block Data
```

Appendix A

UO Constants Tables

A.1 Layers

```
0x01 - One handed weapon
0x02 - Two handed weapon, shield, or misc.
0x03 - Shoes
0x04 - Pants
0x05 - Shirt
0x06 - Helm/Hat
0x07 - Gloves
0x08 - Ring
0x09 - Unused
0x0A - Neck
0x0B - Hair
0x0C - Waist (half apron)
0x0D - Torso (inner) (chest armor)
0x0E - Bracelet
0x0F - Unused (backpack, but backpacks go to 0x15)
0x10 - Facial Hair
0x11 - Torso (middle) (sircoat, tunic, full apron, sash)
0x12 - Earrings
0x13 - Arms
```

```
0x14 - Back (cloak)
0x15 - Backpack
0x16 - Torso (outer) (robe)
0x17 - Legs (outer) (skirt/kilt)
0x18 - Legs (inner) (leg armor)
0x19 - Mount (horse, ostard, etc)
0x1A - NPC Buy Restock container
0x1B - NPC Buy no restock container
0x1C - NPC Sell container
0x1D - PC Bank Box
```

A.2   Direction

```
0x00 - North
0x01 - Northeast
0x02 - East
0x03 - Southeast
0x04 - South
0x05 - Southwest
0x06 - West
0x07 - Northwest
If running, dir = dir|0x80  (so dirs become 0x80, 0x81, etc_)
```

A.3   Speech Types

```
0x00 - Regular
0x01 - Broadcast
0x02 - Emote (adds *'s as part of text)
0x06 - System
0x08 - Whisper
0x09 - Yell
```

A.4 Speech Fonts

```
0x0000 - Bold Text
0x0001 - Text with shadow
0x0002 - Bold+Shadow
0x0003 - Normal
0x0004 - Gothic
0x0005 - Italic Script
0x0006 - Small Dark Letters
0x0007 - Colorful Font (Buggy?)
0x0008 - Rune font (Only use capital letters with this!)
0x0009 - Small Light Letters
```

A.5 Skill List

```
(1 indexed version.  Some messages use it 0 indexed)
1 - Alchemy
2 - Anatomy
3 - Animal Lore
4 - Item ID (Appraise)
5 - Arms Lore
6 - Parrying (Battle Defense)
7 - Begging
8 - Blacksmithing
9 - Bowcraft
10 - Peacemaking (Calming)
11 - Camping
12 - Carpentry
13 - Cartography
14 - Cooking
15 - Detect Hidden
16 - Enticement
17 - Evaluate Intelligence
18 - Healing
19 - Fishing
```

```
20 - Forensic Evaluation
21 - Herding
22 - Hiding
23 - Provocation
24 - Inscription
25 - Lockpicking
26 - Magery
27 - Magic Resistance
28 - Tactics
29 - Snooping
30 - Musicianship
31 - Poisoning
32 - Archery
33 - Spirit Speak
34 - Stealing
35 - Tailoring
36 - Animal Taming (Taming)
37 - Taste Identification
38 - Tinkering
39 - Tracking
40 - Veterinary
41 - Swordsmanship
42 - Mace Fighting
43 - Fencing
44 - Wrestling
45 - Lumberjacking
46 - Mining
47 - Meditation
48 - Stealth
49 - Remove Trap
50 - Necromancy
```

Appendix B

GNU Free Documentation License

Version 1.1, March 2000

Copyright © 2000 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU

61

General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

B.1 Applicability and Definitions

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose

contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

B.2   Verbatim Copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

B.3   Copying in Quantity

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

B.4   Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document,

thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

- State on the Title page the name of the publisher of the Modified Version, as the publisher.

- Preserve all the copyright notices of the Document.

- Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

- Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

- Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

- Include an unaltered copy of this License.

- Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

- Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations

given in the Document for previous versions it was based on. These may be
placed in the "History" section. You may omit a network location for a work
that was published at least four years before the Document itself, or if the
original publisher of the version it refers to gives permission.

- In any section entitled "Acknowledgements" or "Dedications", preserve the
  section's title, and preserve in the section all the substance and tone of each
  of the contributor acknowledgements and/or dedications given therein.

- Preserve all the Invariant Sections of the Document, unaltered in their text
  and in their titles. Section numbers or the equivalent are not considered part
  of the section titles.

- Delete any section entitled "Endorsements". Such a section may not be in-
  cluded in the Modified Version.

- Do not retitle any existing section as "Endorsements" or to conflict in title
  with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that
qualify as Secondary Sections and contain no material copied from the Document,
you may at your option designate some or all of these sections as invariant. To
do this, add their titles to the list of Invariant Sections in the Modified Version's
license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing
but endorsements of your Modified Version by various parties – for example, state-
ments of peer review or that the text has been approved by an organization as the
authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage
of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the
Modified Version. Only one passage of Front-Cover Text and one of Back-Cover
Text may be added by (or through arrangements made by) any one entity. If the
Document already includes a cover text for the same cover, previously added by
you or by arrangement made by the same entity you are acting on behalf of, you
may not add another; but you may replace the old one, on explicit permission from
the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

B.5 Combining Documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

B.6 Collections of Documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the

extracted document, and follow this License in all other respects regarding verbatim copying of that document.

B.7   Aggregation With Independent Works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

B.8   Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

B.9   Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sub-

license or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

B.10 Future Revisions of This License

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.