

Format of .utl Files (inferred)

Eskarina of Morningthaw/Coldeve

February 17, 2022

Contents

1	Introduction	2
2	Overall Format	3
3	Actions	3
4	Requirements (Menu-ordering)	4
4.1	Enable/Disable (9999; rule prefixes ‘ON:’ and ‘NO:’)	4
4.2	Armor Type Similar Color (15; ArmorColorLike)	4
4.3	Base Skill Range (1004; BaseSkillRange)	6
4.4	Buffed Double Value Key \geq (2005; DKeyBuffedGE)	8
4.5	Buffed Long Value Key \geq (2003; LKeyBuffedGE)	8
4.6	Calced Buffed Median Damage \geq (2000; CalcedBuffedMedianDmgGE)	8
4.7	Calced Buffed Missile Damage \geq (2001; CalcedBuffedMissileDmgGE)	8
4.8	Calced Buffed Tinked Dmg \geq (2006; CalcedBuffedTinkDmgGE)	8
4.9	Calced Buffed Tinked Target (2008; CalcedBuffedTinkTarget)	8
4.10	Calced Total Ratings \geq (2007; CalcedTotalRatingsGE)	9
4.11	Character Buffed Skill \geq (1000; BuffedSkillGE)	9
4.12	Character Level \leq (1003; CharLevelLE)	9
4.13	Character Level Key \geq (1002; CharLevelGE)	9
4.14	Damage Percentage \geq (6; DmgPercentGE)	9
4.15	Double Value Key \leq (4; DKeyLE)	9
4.16	Double Value Key \geq (5; DKeyGE)	9
4.17	Free Main Pack Slots \geq (1001; EmptyMainPackSlotsGE)	10
4.18	Long Value Key \neq (13; LKeyNE)	10
4.19	Long Value Key \leq (2; LKeyLE)	10
4.20	Long Value Key $=$ (12; LKeyE)	10
4.21	Long Value Key \geq (3; LKeyGE)	10
4.22	Long Value Key Has Flags (11; LKeyFlags)	10
4.23	Minimum Damage \geq (10; MinDmgGE)	10
4.24	ObjectClass (7; ObjClass)	10
4.25	One Similar Color (14; ColorLike)	12

4.26 Slot Exact Palette (17; SlotPalette)	12
4.27 Slot Similar Color (16; SlotColorLike)	12
4.28 Spell Count \geq (8; NSpellsGE)	12
4.29 Spell Match and Count (9; NSpellsRxGE)	13
4.30 Spell Name Match (0; SpellRx)	13
4.31 String Value Match (1; MatchRx)	13
4.32 Grouped Requirement Descriptions	14
4.32.1 Double Value Key Requirements	14
4.32.2 Long Value Key Requirements	15
5 Salvage Rules	18
A Data Tables in Code-Order	21

List of Tables

1 Action codes.	4
2 ArmorTypeAndColorRegion strings.	5
3 Skill codes.	6
4 ObjectClass codes.	11
5 String Value Key field codes.	13
6 Double Value Key RequirementType codes.	14
7 Double Value Key field codes.	14
8 Long Value Key RequirementType codes.	15
9 Long Value Key field codes.	16
10 SalvageType codes.	19
11 Code-ordered RequirementType codes.	21
12 Menu- & Code-ordered Armor Set codes.	22
13 Code-ordered Double Value Key field codes.	23
14 Code-ordered Long Value Key field codes.	24
15 Code-ordered Skill codes.	27
16 Code-ordered ObjectClass codes.	29
17 Code-ordered String Value Key field codes.	30
18 Code-ordered SalvageType codes.	31

1 Introduction

After creating the `metaf` tool, I later moved on to methodically reverse engineer the `.utl` file format and create the `myutiloctor` tool. The information in this document summarizes my discoveries. I make no promises as to its accuracy, though it does seem quite workable.

In my `.met` and `.nav` file format document, I briefly included a description of the general “VT Table” file format. While there are other VT files that conform to that format, `.utl` files are not among them. They do similarly require sequential reading to consume their contents, though; so if

there's a formatting error at some point, usually all subsequent data cannot be properly read due to its misalignment with formatting expectations.

2 Overall Format

The high-level view of the .utl data format is as follows.

Files are:

```

1 UTL #static string
2 1 #static 1 (maybe number of tables saved in file?)
3 RuleCount #number of loot rules in file
4 [Rules] #multiple lines; iterated to define RuleCount number of loot rules
5 SalvageCombine #static string
6 ByteArrayCount #byte count of entire following salvage section's text, incl. newlines
7 1 #static 1; unknown meaning (default something?)
8 DefaultCombinationText #salvage's Default Combination rule, e.g., 1-6, 7-8, 9, 10
9 SalvageCraftRuleCount #number of specific-salvage-type workmanship-combination rules
10 [SalvageRules] #multiple lines; iterated to define SalvageCraftRuleCount combo rules
11 SalvageValueRuleCount #number of specific-salvage-type value-combination rules
12 [SalvageRules] #multiple lines; iterated to define SalvageValueRuleCount combo rules

```

Rules are:

```

1 RuleName #name of loot rule
2 #always-blank line (?)
3 0;Action;RequirementType;...;RequirementType #static 0 (?); response action code if rule
  matches (e.g., Keep, Sell, etc.); zero or more requirement-type codes
4 KeepNumber #line is only present if Action is "Keep#"; integer number of items to keep
5 [Requirements] #multiple lines; iterated to define all requirements for listed
  requirement type codes, respectively

```

The possible Action codes are listed in [Table 1](#) and RequirementType codes in [Table 11](#). Requirements' data-contents vary in format, as specified in [Section 4](#).

SalvageRules are:

```

1 SalvageType #integer code specifying type of salvage
2 CombinationRule #string; combo rule for this salvage type; e.g., 1-7,8-9,10 or 85000

```

The SalvageRules are strictly split into a list of all workmanship-combo rules followed by a list of all value-combo rules; they are not mixed. Possible SalvageType codes are listed in [Table 10](#). See [Section 5](#) for more on SalvageRules.

3 Actions

Loot rules support five different response actions, listed in [Table 1](#) below. They indicate that any item matching the loot rule should be kept, salvaged, sold, read (e.g., a magic scroll), or kept but only up to a designated item-count cap.

Table 1: Action codes.

<u>Code</u>	<u>ActionType (on menu)</u>	<u>myutilootor</u>
1	Keep	Keep
2	Salvage	Salvage
3	Sell	Sell
4	Read	Read
10	Keep #	Keep # (where # is an integer)

4 Requirements (Menu-ordering)

Loot rules include zero or more Requirements, all of which must be satisfied (except perhaps ‘Enable/Disable’) for a rule to evaluate to true and thereby trigger its selected action in response. Requirements vary in format, as specified in this section. For a RequirementType code summary, see [Table 11](#).

Red coloration of items throughout this document indicates that contacting the server is required for actual evaluation; items in black may or may not require this, depending on whether they are acted-on keys/fields (don’t) or requirement-types (might, depending upon the key/field acted on).

4.1 Enable/Disable (9999; rule prefixes ‘ON:’ and ‘NO:’)

RequirementType code 9999 (in rule’s semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 Status #boolean; true=DISabled or false=ENabled
```

This is the outlier requirement type in that it is unlike any of the others. It merely enables or disables an entire rule if it is present in that rule. Note that ‘true’ means DISable, and ‘false’ means ENable. This requirement may appear anywhere in the requirements list and may even appear multiple times, but note that any single ‘true’ (disable) overrides all other occurrences, regardless of their value. (myutilootor reduces this requirement to a single representation, and in an outlier way as well: it specifies it via the ‘ON:’ (enable) and ‘NO:’ (disable) rule prefixes.)

4.2 Armor Type Similar Color (15; ArmorColorLike)

RequirementType code 15 (in rule’s semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 RedValue #integer in [0,255]
3 GreenValue #integer in [0,255]
4 BlueValue #integer in [0,255]
5 MaxDiffHue #integer in [0,255]
6 MaxDiffSV #double in [0,1]
7 ArmorTypeAndColorRegion #string from specific string set
```

ArmorTypeAndColorRegion may be any of the strings in [Table 2](#) below. (Yes, these exact strings are what actually appear in .utl files, rather than integer codes representing them.)

Table 2: ArmorTypeAndColorRegion strings.

<u>ArmorTypeAndColorRegion (on menu)</u>	<u>myutilootor</u>
None	A_None
Amuli Coat (Chest)	A_Amuli_Coat_Chest
Amuli Coat (Collar/Shoulder)	A_Amuli_Coat_CollarShoulder
Amuli Coat (Arms/Trim)	A_Amuli_Coat_ArmsTrim
Amuli Legs (Base)	A_Amuli_Legs_Base
Amuli Legs (Trim)	A_Amuli_Legs_Trim
Celdon (Base)	A_Celdon_Base
Celdon (Veins)	A_Celdon_Veins
Chiran Coat (Base/Arms)	A_Chiran_Coat_BaseArms
Chiran Coat (Stripes)	A_Chiran_Coat_Stripes
Chiran Legs (Girth)	A_Chiran_Legs_Girth
Chiran Legs (Legs)	A_Chiran_Legs_Legs
Chiran Legs (Trim)	A_Chiran_Legs_Trim
Chiran Helm (Horns)	A_Chiran_Helm_Horns
Chiran Helm (Base)	A_Chiran_Helm_Base
Haebrean BP (Chest) *	A_Haebrean_BP_Chest
Haebrean BP (Ornaments)	A_Haebrean_BP_Ornaments
Haebrean BP (Trim)	A_Haebrean_BP_Trim
Haebrean Girth (Base) *	A_Haebrean_Girth_Base
Haebrean Girth (Belt/Scales)	A_Haebrean_Girth_BeltScales
Haebrean Helm (Base)	A_Haebrean_Helm_Base
Haebrean Helm (Mask)	A_Haebrean_Helm_Mask
Haebrean Pauldrons (Base) *	A_Haebrean_Pauldrons_Base
Haebrean Pauldrons (Ornaments)	A_Haebrean_Pauldrons_Ornaments
Lorica BP (Veins)	A_Lorica_BP_Veins
Lorica BP (Base)	A_Lorica_BP_Base
Lorica BP (Neck/Trim) *	A_Lorica_BP_NeckTrim
Lorica Legs (Base)	A_Lorica_Legs_Base
Lorica Legs (Knees/Belt/Crotch) *	A_Lorica_Legs_KneesBeltCrotch
Lorica Legs (Legs) *	A_Lorica_Legs_Legs
Nariyid BP (Circle/Lines)	A_Nariyid_BP_CircleLines
Nariyid BP (Base)	A_Nariyid_BP_Base
Nariyid BP (Shoulders)	A_Nariyid_BP_Shoulders
Nariyid Girth (Base) *	A_Nariyid_Girth_Base
Nariyid Girth (Belt/Lines)	A_Nariyid_Girth_BeltLines
Nariyid Girth (Ornaments)	A_Nariyid_Girth_Ornaments
Nariyid Sleeves (Shoulders)	A_Nariyid_Sleeves_Shoulders
Nariyid Sleeves (Upper Arm)	A_Nariyid_Sleeves_UpperArm
Nariyid Sleeves (Lower Arm)	A_Nariyid_Sleeves_LowerArm
Olthoi BP (Base)	A_Olthoi_BP_Base
Olthoi BP (Veins)	A_Olthoi_BP_Veins

(continues on next page)

Table 2: (continued)

<u>ArmorTypeAndColorRegion (on menu)</u>	<u>myutilootor</u>
Olthoi Alduressa Legs (Girth: Base)	A_Olthoi_Alduressa_Legs_GirthBase
Olthoi Alduressa Legs (Girth: Lines)	A_Olthoi_Alduressa_Legs_GirthLines
Olthoi Alduressa Legs (Legs: Lines)	A_Olthoi_Alduressa_Legs_LegsLines
Olthoi Amuli Coat (Base) *	A_Olthoi_Amuli_Coat_Base
Olthoi Amuli Coat (Trim)	A_Olthoi_Amuli_Coat_Trim
Olthoi Amuli Coat (Shoulders)	A_Olthoi_Amuli_Coat_Shoulders
Olthoi Amuli Legs (Trim)	A_Olthoi_Amuli_Legs_Trim
Olthoi Koujia Kabuton (Base)	A_Olthoi_Koujia_Kabuton_Base
Olthoi Koujia Kabuton (Horns)	A_Olthoi_Koujia_Kabuton_Horns
Olthoi Koujia Legs (Base)	A_Olthoi_Koujia_Legs_Base
Olthoi Koujia Legs (Sides/Shins)	A_Olthoi_Koujia_Legs_SidesShins
Scalemail Cuirass (Base)	A_Scalemail_Cuirass_Base
Scalemail Cuirass (Bumps)	A_Scalemail_Cuirass_Bumps
Scalemail Cuirass (Belt)	A_Scalemail_Cuirass_Belt
Tenassa Legs (Line at Side)	A_Tenassa_Legs_LineAtSide
Tenassa Legs (Base)	A_Tenassa_Legs_Base
Tenassa Legs (Hilight)	A_Tenassa_Legs_Highlight
Tenassa BP (Shoulders)	A_Tenassa_BP_Shoulders
Tenassa BP (Base)	A_Tenassa_BP_Base
Yoroi Cuirass (Base)	A_Yoroi_Cuirass_Base
Yoroi Cuirass (Belt)	A_Yoroi_Cuirass_Belt
Yoroi Girth (Base)	A_Yoroi_Girth_Base
Yoroi Girth (Belt)	A_Yoroi_Girth_Belt

4.3 Base Skill Range (1004; BaseSkillRange)

RequirementType code 1004 (in rule's semicolon-delimited list). Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 ActsOnID #integer code for a skill type
3 MinimumBaseSkill #integer
4 MaximumBaseSkill #integer

```

ActsOnID may be any of the codes in [Table 3](#) below.

Table 3: Skill codes.

(See [Table 15](#) for code-ordering of these entries.)

<u>Code</u>	<u>Skill (on menu)</u>	<u>myutilootor</u>
38	Alchemy	S_Alchemy
14	ArcaneLore	S_ArcaneLore
29	ArmorTinkering	S_ArmorTinkering
27	AssessCreature	S_AssessCreature
19	AssessPerson	S_AssessPerson

(continues on next page)

Table 3: (continued)

<u>Code</u>	<u>Skill (on menu)</u>	<u>myutilootor</u>
1	Axe (deprecated)	S_Axe
2	Bow (deprecated)	S_Bow
39	Cooking	S_Cooking
31	CreatureEnchantment	S_CreatureEnchantment
3	Crossbow (deprecated)	S_Crossbow
4	Dagger (deprecated)	S_Dagger
20	Deception	S_Deception
52	DirtyFighting	S_DirtyFighting
49	DualWield	S_DualWield
46	FinesseWeapons	S_FinesseWeapons
37	Fletching	S_Fletching
42	Gearcraft (deprecated)	S_Gearcraft
21	Healing	S_Healing
44	HeavyWeapons	S_HeavyWeapons
32	ItemEnchantment	S_ItemEnchantment
18	ItemTinkering	S_ItemTinkering
22	Jump	S_Jump
35	Leadership	S_Leadership
33	LifeMagic	S_LifeMagic
45	LightWeapons	S_LightWeapons
23	Lockpick	S_Lockpick
36	Loyalty	S_Loyalty
5	Mace (deprecated)	S_Mace
15	MagicDefense	S_MagicDefense
30	MagicItemTinkering	S_MagicItemTinkering
16	ManaConversion	S_ManaConversion
6	MeleeDefense	S_MeleeDefense
7	MissileDefense	S_MissileDefense
47	MissileWeapons	S_MissileWeapons
50	Recklessness	S_Recklessness
24	Run	S_Run
40	Salvaging	S_Salvaging
48	Shield	S_Shield
51	SneakAttack	S_SneakAttack
9	Spear (deprecated)	S_Spear
10	Staff (deprecated)	S_Staff
54	Summoning	S_Summoning
11	Sword (deprecated)	S_Sword
12	ThrownWeapons (deprecated)	S_ThrownWeapons
41	TwoHandedCombat	S_TwoHandedCombat
13	Unarmed (deprecated)	S_Unarmed
43	VoidMagic	S_VoidMagic

(continues on next page)

Table 3: (continued)

<u>Code</u>	<u>Skill (on menu)</u>	<u>myutilootor</u>
34	WarMagic	S_WarMagic
28	WeaponTinkering	S_WeaponTinkering

4.4 Buffed Double Value Key >= (2005; DKeyBuffedGE)

RequirementType code 2005 (in rule's semicolon-delimited list). Hits server because all valid acted-on keys/fields require that. See [Section 4.32.1](#) for more.

4.5 Buffed Long Value Key >= (2003; LKeyBuffedGE)

RequirementType code 2003 (in rule's semicolon-delimited list). Hits server because all valid acted-on keys/fields require that. See [Section 4.32.2](#) for more.

4.6 Calced Buffed Median Damage >= (2000; CalcedBuffedMedianDmgGE)

RequirementType code 2000 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CalcedBuffedMedian #double-type value
```

4.7 Calced Buffed Missile Damage >= (2001; CalcedBuffedMissileDmgGE)

RequirementType code 2001 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CalcedBuffedMissile #double-type value
```

4.8 Calced Buffed Tinked Dmg >= (2006; CalcedBuffedTinkDmgGE)

RequirementType code 2006 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CalcedBuffedTinked #double-type value
```

4.9 Calced Buffed Tinked Target (2008; CalcedBuffedTinkTarget)

RequirementType code 2008 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 TargetCalcedBuffed #double
3 TargetBuffedMelee #double
4 TargetBuffedAttackBonus #double
```


4.10 Calced Total Ratings >= (2007; CalcedTotalRatingsGE)

RequirementType code 2007 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CalcedTotalRatings #double-type value
```

4.11 Character Buffed Skill >= (1000; BuffedSkillGE)

RequirementType code 1000 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 BuffedSkillValue #integer (?)
3 ActsOnID #integer code for a skill type
```

ActsOnID may be any of the codes in [Table 3](#) (or [Table 15](#)).

4.12 Character Level <= (1003; CharLevelLE)

RequirementType code 1003 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CharacterLevel #integer
```

4.13 Character Level Key >= (1002; CharLevelGE)

RequirementType code 1002 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 CharacterLevel #integer
```

4.14 Damage Percentage >= (6; DmgPercentGE)

RequirementType code 6 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 DamagePercent #double-type value
```

4.15 Double Value Key <= (4; DKeyLE)

RequirementType code 4 (in rule's semicolon-delimited list). See [Section 4.32.1](#) for more.

4.16 Double Value Key >= (5; DKeyGE)

RequirementType code 5 (in rule's semicolon-delimited list). See [Section 4.32.1](#) for more.

4.17 Free Main Pack Slots \geq (1001; EmptyMainPackSlotsGE)

RequirementType code 1001 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 EmptySlots #integer
```

4.18 Long Value Key \neq (13; LKeyNE)

RequirementType code 13 (in rule's semicolon-delimited list). See [Section 4.32.2](#) for more.

4.19 Long Value Key \leq (2; LKeyLE)

RequirementType code 2 (in rule's semicolon-delimited list). See [Section 4.32.2](#) for more.

4.20 Long Value Key $=$ (12; LKeyE)

RequirementType code 12 (in rule's semicolon-delimited list). See [Section 4.32.2](#) for more.

4.21 Long Value Key \geq (3; LKeyGE)

RequirementType code 3 (in rule's semicolon-delimited list). See [Section 4.32.2](#) for more.

4.22 Long Value Key Has Flags (11; LKeyFlags)

RequirementType code 11 (in rule's semicolon-delimited list). See [Section 4.32.2](#) for more.

4.23 Minimum Damage \geq (10; MinDmgGE)

RequirementType code 10 (in rule's semicolon-delimited list). Hits server. Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 MinimumDamage #double-type value
```

4.24 ObjectClass (7; ObjClass)

RequirementType code 7 (in rule's semicolon-delimited list). Content format:

```
1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 ObjectClassID #integer code
```

ObjectClassID may be any of the codes in [Table 4](#) below.

Table 4: ObjectClass codes.

(See [Table 16](#) for code-ordering of these entries.)

<u>Code</u>	<u>ObjectClass (on menu)</u>	<u>myutilootor</u>
2	Armor	C_Armor
19	BaseAlchemy	C_BaseAlchemy
18	BaseCooking	C_BaseCooking
20	BaseFletching	C_BaseFletching
33	Book	C_Book
32	Bundle	C_Bundle
3	Clothing	C_Clothing
10	Container	C_Container
27	Corpse	C_Corpse
22	CraftedAlchemy	C_CraftedAlchemy
21	CraftedCooking	C_CraftedCooking
23	CraftedFletching	C_CraftedFletching
26	Door	C_Door
38	Foci	C_Foci
6	Food	C_Food
11	Gem	C_Gem
29	HealingKit	C_HealingKit
36	Housing	C_Housing
4	Jewelry	C_Jewelry
34	Journal	C_Journal
13	Key	C_Key
28	Lifestone	C_Lifestone
30	Lockpick	C_Lockpick
16	ManaStone	C_ManaStone
1	MeleeWeapon	C_MeleeWeapon
8	Misc	C_Misc
9	MissileWeapon	C_MissileWeapon
7	Money	C_Money
5	Monster	C_Monster
37	Npc	C_Npc
17	Plant	C_Plant
24	Player	C_Player
14	Portal	C_Portal
39	Salvage	C_Salvage
42	Scroll	C_Scroll
41	Services	C_Services
35	Sign	C_Sign
12	SpellComponent	C_SpellComponent
15	TradeNote	C_TradeNote
0	Unknown	C_Unknown

(continues on next page)

Table 4: (continued)

<u>Code</u>	<u>ObjectClass (on menu)</u>	<u>myutilootor</u>
40	Ust	C_Ust
25	Vendor	C_Vendor
31	WandStaffOrb	C_WandStaffOrb

4.25 One Similar Color (14; ColorLike)

RequirementType code 14 (in rule's semicolon-delimited list). Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 RedValue #integer in [0,255]
3 GreenValue #integer in [0,255]
4 BlueValue #integer in [0,255]
5 MaxDiffHue #integer in [0,255]
6 MaxDiffSV #double in [0,1]
```

4.26 Slot Exact Palette (17; SlotPalette)

RequirementType code 17 (in rule's semicolon-delimited list). Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 PaletteEntryNumber #integer (?)
3 PaletteID #integer
```

4.27 Slot Similar Color (16; SlotColorLike)

RequirementType code 16 (in rule's semicolon-delimited list). Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 RedValue #integer in [0,255]
3 GreenValue #integer in [0,255]
4 BlueValue #integer in [0,255]
5 MaxDiffHue #integer in [0,255]
6 MaxDiffSV #double in [0,1]
7 PaletteEntryNumber #integer (?)
```

4.28 Spell Count >= (8; NSpellsGE)

RequirementType code 8 (in rule's semicolon-delimited list). Hits server. Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 SpellCount #integer
```

4.29 Spell Match and Count (9; NSpellsRxGE)

RequirementType code 9 (in rule's semicolon-delimited list). Hits server. Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 MatchPattern #string; regular expression to match against spell names
3 DoNotMatchPattern #string; regular expression to not-match against spell names
4 MinimumMatchCount #integer; must match at least this many spells

```

4.30 Spell Name Match (0; SpellRx)

RequirementType code 0 (in rule's semicolon-delimited list). Hits server. Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 MatchPattern #string; regular expression to match against spell names

```

4.31 String Value Match (1; MatchRx)

RequirementType code 1 (in rule's semicolon-delimited list). Content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 ActsOnID #integer code for a string field
3 MatchPattern #string; regular expression to match against string field contents

```

ActsOnID may be any of the codes in [Table 5](#) below.

Table 5: String Value Key field codes.
(See [Table 17](#) for code-ordering of these entries.)

Code	StringValueKey (on menu)	myutilootor
43	DateBorn	M_DateBorn
10	FellowshipName	M_FellowshipName
16	FullDescription	M_FullDescription
40	ImbuedBy	M_ImbuedBy
8	InscribedBy	M_InscribedBy
7	Inscription	M_Inscription
39	LastTinkeredBy	M_LastTinkeredBy
21	MonarchName	M_MonarchName
1	Name	M_Name
25	OnlyActivatedBy	M_OnlyActivatedBy
35	Patron	M_Patron
38	PortalDestination	M_PortalDestination
184549376	SecondaryName	M_SecondaryName
15	SimpleDescription	M_SimpleDescription
5	Title	M_Title
14	UsageInstructions	M_UsageInstructions

4.32 Grouped Requirement Descriptions

There are a couple of Requirement groups, the members of which are very similar to one another. For brevity, this section describes two such groups, rather than repeating the same information for each member Requirement.

4.32.1 Double Value Key Requirements

A ‘double’ key refers to a 64-bit ‘floating point’-type (decimal value) data field in an object’s data structure. Since there are several Requirement variants involving ‘double’ fields, they’re all grouped together in this section. With the exception of ‘Buffed Double Value Key >=’, they act on the same set of possible data fields, which are listed in [Table 7](#) below. ‘Buffed Double Value Key >=’ only acts on the four fields marked with asterisks.

All the Double Value Key Requirement variants have the same content format with only their RequirementType codes differing (in the rule’s semicolon-delimited requirements list), as shown in [Table 6](#) below.

Table 6: Double Value Key RequirementType codes.

<u>Code</u>	<u>RequirementType (on menu)</u>	<u>myutilootor</u>
4	Double Value Key <=	DKeyLE
5	Double Value Key >=	DKeyGE
2005	Buffed Double Value Key >=	DKeyBuffedGE

All Double Value Key RequirementTypes’ data-content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 DoubleValue #double value to compare the double field contents to
3 ActsOnID #integer code for a double field

```

Table 7: Double Value Key field codes.

(See [Table 13](#) for code-ordering of these entries.)

<u>Code</u>	<u>DoubleValueKey (on menu)</u>	<u>myutilootor</u>
167772163	AcidProt	D_AcidProt
167772168	ApproachDistance	D_ApproachDistance
* 167772172	AttackBonus	D_AttackBonus
167772162	BludgeonProt	D_BludgeonProt
167772166	ColdProt	D_ColdProt
167772174	DamageBonus	D_DamageBonus
* 152	ElementalDamageVersus	D_ElementalDamageVersus
167772165	FireProt	D_FireProt
167772167	Heading	D_Heading
100	HealingKitRestoreBonus	D_HealingKitRestoreBonus
167772164	LightningProt	D_LightningProt

(continues on next page)

Table 7: (continued)

<u>Code</u>	<u>DoubleValueKey (on menu)</u>	<u>myutilootor</u>
150	MagicDBonus	D_MagicDBonus
* 144	ManaCBonus	D_ManaCBonus
5	ManaRateOfChange	D_ManaRateOfChange
137	ManaStoneChanceDestruct	D_ManaStoneChanceDestruct
87	ManaTransferEfficiency	D_ManaTransferEfficiency
* 29	MeleeDefenseBonus	D_MeleeDefenseBonus
149	MissileDBonus	D_MissileDBonus
167772161	PierceProt	D_PierceProt
167772173	Range	D_Range
167772169	SalvageWorkmanship	D_SalvageWorkmanship
167772170	Scale	D_Scale
167772160	SlashProt	D_SlashProt
167772171	Variance	D_Variance

4.32.2 Long Value Key Requirements

A ‘long’ key refers to a ‘long integer’-type data field in an object’s data structure. Since there are several Requirement variants involving ‘long’ fields, they’re all grouped together in this section. With the exception of ‘Buffed Long Value Key >=’, they all act on the same set of possible data fields, which are listed in [Table 9](#) below. ‘Buffed Long Value Key >=’ only acts on the two fields marked with asterisks.

All the Long Value Key Requirement variants have the same content format with only their RequirementType codes differing (in the rule’s semicolon-delimited requirements list), as shown in [Table 8](#) below.

Table 8: Long Value Key RequirementType codes.

<u>Code</u>	<u>RequirementType (on menu)</u>	<u>myutilootor</u>
2	Long Value Key <=	LKeyLE
3	Long Value Key >=	LKeyGE
11	Long Value Key Has Flags	LKeyFlags
12	Long Value Key ==	LKeyE
13	Long Value Key !=	LKeyNE
2003	Buffed Long Value Key >=	LKeyBuffedGE

All Long Value Key RequirementTypes’ data-content format:

```

1 ByteArrayCount #integer; byte count from next line through end of req, incl. newlines
2 LongValue #integer value to compare the long field contents to
3 ActsOnID #integer code for a long field

```

Table 9: Long Value Key field codes.
(See [Table 14](#) for code-ordering of these entries.)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutiloctor</u>
176	ActivationReqSkillId	L_ActivationReqSkillId
218103848	ActiveSpellCount	L_ActiveSpellCount
90	AffectsVitalAmt	L_AffectsVitalAmt
89	AffectsVitalId	L_AffectsVitalId
125	Age	L_Age
* 28	ArmorLevel	L_ArmorLevel
265	ArmorSetID	L_ArmorSetID
218103816	AssociatedSpell	L_AssociatedSpell
114	Attuned	L_Attuned
218103835	Behavior	L_Behavior
33	Bonded	L_Bonded
5	Burden	L_Burden
218103834	Category	L_Category
352	CloakChanceType	L_CloakChanceType
218103810	Container	L_Container
167	CooldownSeconds	L_CooldownSeconds
218103821	Coverage	L_Coverage
218103832	CreateFlags1	L_CreateFlags1
218103833	CreateFlags2	L_CreateFlags2
25	CreatureLevel	L_CreatureLevel
374	CritDamRating	L_CritDamRating
375	CritDamResistRating	L_CritDamResistRating
372	CritRating	L_CritRating
373	CritResistRating	L_CritResistRating
107	CurrentMana	L_CurrentMana
218103841	DamageType	L_DamageType
370	DamRating	L_DamRating
371	DamResistRating	L_DamResistRating
98	DateOfBirth	L_DateOfBirth
43	Deaths	L_Deaths
172	DescriptionFormat	L_DescriptionFormat
204	ElementalDmgBonus	L_ElementalDmgBonus
218103822	EquipableSlots	L_EquipableSlots
10	EquippedSlots	L_EquippedSlots
218103840	EquipSkill	L_EquipSkill
218103823	EquipType	L_EquipType
192	FishingSkill	L_FishingSkill
218103831	Flags	L_Flags
177	GemSettingQty	L_GemSettingQty
178	GemSettingType	L_GemSettingType

(continues on next page)

Table 9: (continued)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutilootor</u>
113	Gender	L_Gender
376	HealBoostRating	L_HealBoostRating
188	Heritage	L_Heritage
218103828	HookMask	L_HookMask
218103829	HookType	L_HookType
218103827	HouseOwner	L_HouseOwner
218103809	Icon	L_Icon
218103824	IconOutline	L_IconOutline
218103849	IconOverlay	L_IconOverlay
218103850	IconUnderlay	L_IconUnderlay
179	Imbued	L_Imbued
319	ItemMaxLevel	L_ItemMaxLevel
218103812	ItemSlots	L_ItemSlots
193	KeysHeld	L_KeysHeld
218103811	Landblock	L_Landblock
38	LockpickDifficulty	L_LockpickDifficulty
88	LockpickSkillBonus	L_LockpickSkillBonus
109	LoreRequirement	L_LoreRequirement
218103836	MagicDef	L_MagicDef
117	ManaCost	L_ManaCost
131	Material	L_Material
* 218103842	MaxDamage	L_MaxDamage
108	MaximumMana	L_MaximumMana
87	MaxLevelRestrict	L_MaxLevelRestrict
86	MinLevelRestrict	L_MinLevelRestrict
218103825	MissileType	L_MissileType
218103830	Model	L_Model
218103820	Monarch	L_Monarch
35	NumberFollowers	L_NumberFollowers
170	NumberItemsSalvaged	L_NumberItemsSalvaged
171	NumberTimesTinkered	L_NumberTimesTinkered
218103813	PackSlots	L_PackSlots
175	PagesTotal	L_PagesTotal
174	PagesUsed	L_PagesUsed
218103847	PhysicsDataFlags	L_PhysicsDataFlags
111	PortalRestrictions	L_PortalRestrictions
30	Rank	L_Rank
110	RankRequirement	L_RankRequirement
17	RareId	L_RareId
26	RestrictedToToD	L_RestrictedToToD
24	SkillCreditsAvail	L_SkillCreditsAvail
115	SkillLevelReq	L_SkillLevelReq

(continues on next page)

Table 9: (continued)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutilootor</u>
166	SlayerSpecies	L_SlayerSpecies
218103817	Slot	L_Slot
218103837	SpecialProps	L_SpecialProps
2	Species	L_Species
218103838	SpellCount	L_SpellCount
106	Spellcraft	L_Spellcraft
218103814	StackCount	L_StackCount
218103815	StackMax	L_StackMax
367	SummoningGemBuffed	L_SummoningGemBuffed
369	SummoningGemLevel	L_SummoningGemLevel
20	TotalValue	L_TotalValue
218103808	Type	L_Type
36	Unenchatable	L_Unenchatable
218103843	Unknown10	L_Unknown10
218103844	Unknown100000	L_Unknown100000
218103845	Unknown800000	L_Unknown800000
218103845	Unknown8000000	L_Unknown8000000
218103826	UsageMask	L_UsageMask
92	UsesRemaining	L_UsesRemaining
91	UsesTotal	L_UsesTotal
19	Value	L_Value
379	VitalityRating	L_VitalityRating
45	WandElemDmgType	L_WandElemDmgType
353	WeaponMasteryCategory	L_WeaponMasteryCategory
218103839	WeapSpeed	L_WeapSpeed
218103818	Wielder	L_Wielder
218103819	WieldingSlot	L_WieldingSlot
159	WieldReqAttribute	L_WieldReqAttribute
158	WieldReqType	L_WieldReqType
160	WieldReqValue	L_WieldReqValue
105	Workmanship	L_Workmanship
129	XPFforVPReduction	L_XPFforVPReduction

5 Salvage Rules

As noted in [Section 2](#), salvage rules come in two main flavors: all workmanship-based salvage combinations are listed, followed by all value-based salvage combinations being listed. The Classic Loot Editor interface does not facilitate any method of specifying a value-based rule *without* also including the selected salvage type in the workmanship-based rule list. (Even if you leave that field blank, it is still in that list.)

It is important to note that there are some messy behaviors to the Classic Loot Editor that may lead to residual data sticking around in the .utl file even if a salvage rule has been deleted!

Specifically, value-based data may not get deleted, even though it should be. (Thus, you may see a file containing a value-based salvage rule for a salvage type that does *not* have a workmanship-based salvage rule list entry. This should not ever happen if the loot editor behaved according to the principle of least surprise. If you play with the application a bit and watch how the file data changes, it's clear that this behavior was not intended. `myutilootor` imposes a more rational behavior on this chaos, issuing warnings where warranted.)

SalvageType may be any of the codes in [Table 10](#) below, which vary from 1 (Ceramic) through 77 (Teak), except for 3, 9, 56, 65, and 72 (not materials).

Table 10: SalvageType codes.
(1 through 77, omitting 3, 9, 56, 65, and 72)
(See [Table 18](#) for code-ordering of these entries.)

<u>Code</u>	<u>SalvageType (on menu)</u>	<u>myutilootor</u>
10	Agate	V_Agate
66	Alabaster	V_Alabaster
11	Amber	V_Amber
12	Amethyst	V_Amethyst
13	Aquamarine	V_Aquamarine
53	Armoredillo Hide	V_ArmoredilloHide
14	Azurite	V_Azurite
15	Black Garnet	V_BlackGarnet
16	Black Opal	V_BlackOpal
17	Bloodstone	V_Bloodstone
57	Brass	V_Brass
58	Bronze	V_Bronze
18	Carnelian	V_Carnelian
1	Ceramic	V_Ceramic
19	Citrine	V_Citrine
59	Copper	V_Copper
20	Diamond	V_Diamond
73	Ebony	V_Ebony
21	Emerald	V_Emerald
22	Fire Opal	V_FireOpal
60	Gold	V_Gold
67	Granite	V_Granite
23	Green Garnet	V_GreenGarnet
24	Green Jade	V_GreenJade
54	Gromnie Hide	V_GromnieHide
25	Hematite	V_Hematite
26	Imperial Topaz	V_ImperialTopaz
61	Iron	V_Iron
51	Ivory	V_Ivory
27	Jet	V_Jet

(continues on next page)

Table 10: (continued)

<u>Code</u>	<u>SalvageType (on menu)</u>	<u>myutilootor</u>
28	Lapis Lazuli	V_LapisLazuli
29	Lavender Jade	V_LavenderJade
52	Leather	V_Leather
4	Linen	V_Linen
74	Mahogany	V_Mahogany
30	Malachite	V_Malachite
68	Marble	V_Marble
31	Moonstone	V_Moonstone
75	Oak	V_Oak
69	Obsidian	V_Obsidian
32	Onyx	V_Onyx
33	Opal	V_Opal
34	Peridot	V_Peridot
76	Pine	V_Pine
2	Porcelain	V_Porcelain
62	Pyreal	V_Pyreal
35	Red Garnet	V_RedGarnet
36	Red Jade	V_RedJade
55	Reed Shark Hide	V_ReedSharkHide
37	Rose Quartz	V_RoseQuartz
38	Ruby	V_Ruby
70	Sandstone	V_Sandstone
39	Sapphire	V_Sapphire
5	Satin	V_Satin
71	Serpentine	V_Serpentine
6	Silk	V_Silk
63	Silver	V_Silver
40	Smokey Quartz	V_SmokeyQuartz
64	Steel	V_Steel
41	Sunstone	V_Sunstone
77	Teak	V_Teak
42	Tiger Eye	V_TigerEye
43	Tourmaline	V_Tourmaline
44	Turquoise	V_Turquoise
7	Velvet	V_Velvet
45	White Jade	V_WhiteJade
46	White Quartz	V_WhiteQuartz
47	White Sapphire	V_WhiteSapphire
8	Wool	V_Wool
48	Yellow Garnet	V_YellowGarnet
49	Yellow Topaz	V_YellowTopaz
50	Zircon	V_Zircon

A Data Tables in Code-Order

Table 11: Code-ordered RequirementType codes.
(See [Table of Contents](#) for menu-ordering of these entries.)

<u>Code</u>	<u>RequirementType (on menu)</u>	<u>myutilootor</u>	<u>Details</u>
0	Spell Name Match	SpellRx	Section 4.30
1	String Value Match	MatchRx	Section 4.31
2	Long Value Key <=	LKeyLE	Section 4.19
3	Long Value Key >=	LKeyGE	Section 4.21
4	Double Value Key <=	DKeyLE	Section 4.15
5	Double Value Key >=	DKeyGE	Section 4.16
6	Damage Percentage >=	DmgPercentGE	Section 4.14
7	ObjectClass	ObjClass	Section 4.24
8	Spell Count >=	NSpellsGE	Section 4.28
9	Spell Match and Count	NSpellsRxGE	Section 4.29
10	Minimum Damage >=	MinDmgGE	Section 4.23
11	Long Value Key Has Flags (?)	LKeyFlags	Section 4.22
12	Long Value Key ==	LKeyE	Section 4.20
13	Long Value Key !=	LKeyNE	Section 4.18
14	One Similar Color	ColorLike	Section 4.25
15	Armor Type Similar Color	ArmorColorLike	Section 4.2
16	Slot Similar Color	SlotColorLike	Section 4.27
17	Slot Exact Palette	SlotPalette	Section 4.26
1000	Character Buffed Skill >=	BuffedSkillGE	Section 4.11
1001	Free Main Pack Slots >=	EmptyMainPackSlotsGE	Section 4.17
1002	Character Level >=	CharLevelGE	Section 4.13
1003	Character Level <=	CharLevelLE	Section 4.12
1004	Base Skill Range	BaseSkillRange	Section 4.3
2000	Calced Buffed Median Damage >=	CalcedBuffedMedianDmgGE	Section 4.6
2001	Calced Buffed Missile Damage >=	CalcedBuffedMissileDmgGE	Section 4.7
2003	Buffed Long Value Key >=	LKeyBuffedGE	Section 4.5
2005	Buffed Double Value Key >=	DKeyBuffedGE	Section 4.4
2006	Calced Buffed Tinked Dmg >=	CalcedBuffedTinkDmgGE	Section 4.8
2007	Calced Total Ratings >=	CalcedTotalRatingsGE	Section 4.10
2008	Calced Buffed Tinked Target (?)	CalcedBuffedTinkTarget	Section 4.9
9999	Enable/Disable	N/A (use ON:/NO:)	Section 4.1

Table 12: Menu- & Code-ordered Armor Set codes.

(For much more, see https://asheron.fandom.com/wiki/EQUIPMENT_SET_ID_INT.)(Or, for just cloaks, see <http://acpedia.org/wiki/Talk:Cloaks>.)

<u>Code</u>	<u>Set (on menu)</u>	<u>Code</u>	<u>Set (on menu)</u>
27	Acid Proof	5	Noble Relic
14	Adept's	6	Ancient Relic
6	Ancient Relic	7	Relic Alduressa
15	Archer's	8	Shou-jen
10	Arm, Mind, Heart	9	Empyrean Rings
11	Coat of the Perfect Light	10	Arm, Mind, Heart
28	Cold Proof	11	Coat of the Perfect Light
18	Crafter's	12	Leggings of Perfect Light
30	Dedication	13	Soldier's
16	Defender's	14	Adept's
20	Dexterous	15	Archer's
9	Empyrean Rings	16	Defender's
26	Flame Proof	17	Tinker's
31	Gladiatorial Clothing	18	Crafter's
23	Hardened	19	Hearty
19	Hearty	20	Dexterous
25	Interlocking	21	Wise
12	Leggings of Perfect Light	22	Swift
29	Lightning Proof	23	Hardened
5	Noble Relic	24	Reinforced
32	Protective Clothing	25	Interlocking
24	Reinforced	26	Flame Proof
7	Relic Alduressa	27	Acid Proof
8	Shou-jen	28	Cold Proof
13	Soldier's	29	Lightning Proof
22	Swift	30	Dedication
17	Tinker's	31	Gladiatorial Clothing
21	Wise	32	Protective Clothing

Table 13: Code-ordered Double Value Key field codes.

(See [Table 7](#) for menu-ordering of these entries.)

<u>Code</u>	<u>DoubleValueKey (on menu)</u>	<u>myutilootor</u>
5	ManaRateOfChange	D_ManaRateOfChange
* 29	MeleeDefenseBonus	D_MeleeDefenseBonus
87	ManaTransferEfficiency	D_ManaTransferEfficiency
100	HealingKitRestoreBonus	D_HealingKitRestoreBonus
137	ManaStoneChanceDestruct	D_ManaStoneChanceDestruct
* 144	ManaCBonus	D_ManaCBonus
149	MissileDBonus	D_MissileDBonus
150	MagicDBonus	D_MagicDBonus
* 152	ElementalDamageVersus	D_ElementalDamageVersus
167772160	SlashProt	D_SlashProt
167772161	PierceProt	D_PierceProt
167772162	BludgeonProt	D_BludgeonProt
167772163	AcidProt	D_AcidProt
167772164	LightningProt	D_LightningProt
167772165	FireProt	D_FireProt
167772166	ColdProt	D_ColdProt
167772167	Heading	D_Heading
167772168	ApproachDistance	D_ApproachDistance
167772169	SalvageWorkmanship	D_SalvageWorkmanship
167772170	Scale	D_Scale
167772171	Variance	D_Variance
* 167772172	AttackBonus	D_AttackBonus
167772173	Range	D_Range
167772174	DamageBonus	D_DamageBonus

Table 14: Code-ordered Long Value Key field codes.

(See [Table 9](#) for menu-ordering of these entries.)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutiloctor</u>
2	Species	L_Species
5	Burden	L_Burden
10	EquippedSlots	L_EquippedSlots
17	RareId	L_RareId
19	Value	L_Value
20	TotalValue	L_TotalValue
24	SkillCreditsAvail	L_SkillCreditsAvail
25	CreatureLevel	L_CreatureLevel
26	RestrictedToToD	L_RestrictedToToD
* 28	ArmorLevel	L_ArmorLevel
30	Rank	L_Rank
33	Bonded	L_Bonded
35	NumberFollowers	L_NumberFollowers
36	Unenchantable	L_Unenchantable
38	LockpickDifficulty	L_LockpickDifficulty
43	Deaths	L_Deaths
45	WandElemDmgType	L_WandElemDmgType
86	MinLevelRestrict	L_MinLevelRestrict
87	MaxLevelRestrict	L_MaxLevelRestrict
88	LockpickSkillBonus	L_LockpickSkillBonus
89	AffectsVitalId	L_AffectsVitalId
90	AffectsVitalAmt	L_AffectsVitalAmt
91	UsesTotal	L_UsesTotal
92	UsesRemaining	L_UsesRemaining
98	DateOfBirth	L_DateOfBirth
105	Workmanship	L_Workmanship
106	Spellcraft	L_Spellcraft
107	CurrentMana	L_CurrentMana
108	MaximumMana	L_MaximumMana
109	LoreRequirement	L_LoreRequirement
110	RankRequirement	L_RankRequirement
111	PortalRestrictions	L_PortalRestrictions
113	Gender	L_Gender
114	Attuned	L_Attuned
115	SkillLevelReq	L_SkillLevelReq
117	ManaCost	L_ManaCost
125	Age	L_Age
129	XPForVPReduction	L_XPForVPReduction
131	Material	L_Material
158	WieldReqType	L_WieldReqType

(continues on next page)

Table 14: (continued)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutilootor</u>
159	WieldReqAttribute	L_WieldReqAttribute
160	WieldReqValue	L_WieldReqValue
166	SlayerSpecies	L_SlayerSpecies
167	CooldownSeconds	L_CooldownSeconds
170	NumberItemsSalvaged	L_NumberItemsSalvaged
171	NumberTimesTinkered	L_NumberTimesTinkered
172	DescriptionFormat	L_DescriptionFormat
174	PagesUsed	L_PagesUsed
175	PagesTotal	L_PagesTotal
176	ActivationReqSkillId	L_ActivationReqSkillId
177	GemSettingQty	L_GemSettingQty
178	GemSettingType	L_GemSettingType
179	Imbued	L_Imbued
188	Heritage	L_Heritage
192	FishingSkill	L_FishingSkill
193	KeysHeld	L_KeysHeld
204	ElementalDmgBonus	L_ElementalDmgBonus
265	ArmorSetID	L_ArmorSetID
319	ItemMaxLevel	L_ItemMaxLevel
352	CloakChanceType	L_CloakChanceType
353	WeaponMasteryCategory	L_WeaponMasteryCategory
367	SummoningGemBuffed	L_SummoningGemBuffed
369	SummoningGemLevel	L_SummoningGemLevel
370	DamRating	L_DamRating
371	DamResistRating	L_DamResistRating
372	CritRating	L_CritRating
373	CritResistRating	L_CritResistRating
374	CritDamRating	L_CritDamRating
375	CritDamResistRating	L_CritDamResistRating
376	HealBoostRating	L_HealBoostRating
379	VitalityRating	L_VitalityRating
218103808	Type	L_Type
218103809	Icon	L_Icon
218103810	Container	L_Container
218103811	Landblock	L_Landblock
218103812	ItemSlots	L_ItemSlots
218103813	PackSlots	L_PackSlots
218103814	StackCount	L_StackCount
218103815	StackMax	L_StackMax
218103816	AssociatedSpell	L_AssociatedSpell
218103817	Slot	L_Slot
218103818	Wielder	L_Wielder

(continues on next page)

Table 14: (continued)

<u>Code</u>	<u>LongValueKey (on menu)</u>	<u>myutilootor</u>
218103819	WieldingSlot	L_WieldingSlot
218103820	Monarch	L_Monarch
218103821	Coverage	L_Coverage
218103822	EquipableSlots	L_EquipableSlots
218103823	EquipType	L_EquipType
218103824	IconOutline	L_IconOutline
218103825	MissileType	L_MissileType
218103826	UsageMask	L_UsageMask
218103827	HouseOwner	L_HouseOwner
218103828	HookMask	L_HookMask
218103829	HookType	L_HookType
218103830	Model	L_Model
218103831	Flags	L_Flags
218103832	CreateFlags1	L_CreateFlags1
218103833	CreateFlags2	L_CreateFlags2
218103834	Category	L_Category
218103835	Behavior	L_Behavior
218103836	MagicDef	L_MagicDef
218103837	SpecialProps	L_SpecialProps
218103838	SpellCount	L_SpellCount
218103839	WeapSpeed	L_WeapSpeed
218103840	EquipSkill	L_EquipSkill
218103841	DamageType	L_DamageType
* 218103842	MaxDamage	L_MaxDamage
218103843	Unknown10	L_Unknown10
218103844	Unknown100000	L_Unknown100000
218103845	Unknown800000	L_Unknown800000
218103845	Unknown8000000	L_Unknown8000000
218103847	PhysicsDataFlags	L_PhysicsDataFlags
218103848	ActiveSpellCount	L_ActiveSpellCount
218103849	IconOverlay	L_IconOverlay
218103850	IconUnderlay	L_IconUnderlay

Table 15: Code-ordered Skill codes.
(See [Table 3](#) for menu-ordering of these entries.)

<u>Code</u>	<u>Skill (on menu)</u>	<u>myutilootor</u>
1	Axe (deprecated)	S_Axe
2	Bow (deprecated)	S_Bow
3	Crossbow (deprecated)	S_Crossbow
4	Dagger (deprecated)	S_Dagger
5	Mace (deprecated)	S_Mace
6	MeleeDefense	S_MeleeDefense
7	MissileDefense	S_MissileDefense
9	Spear (deprecated)	S_Spear
10	Staff (deprecated)	S_Staff
11	Sword (deprecated)	S_Sword
12	ThrownWeapons (deprecated)	S_ThrownWeapons
13	Unarmed (deprecated)	S_Unarmed
14	ArcaneLore	S_ArcaneLore
15	MagicDefense	S_MagicDefense
16	ManaConversion	S_ManaConversion
18	ItemTinkering	S_ItemTinkering
19	AssessPerson	S_AssessPerson
20	Deception	S_Deception
21	Healing	S_Healing
22	Jump	S_Jump
23	Lockpick	S_Lockpick
24	Run	S_Run
27	AssessCreature	S_AssessCreature
28	WeaponTinkering	S_WeaponTinkering
29	ArmorTinkering	S_ArmorTinkering
30	MagicItemTinkering	S_MagicItemTinkering
31	CreatureEnchantment	S_CreatureEnchantment
32	ItemEnchantment	S_ItemEnchantment
33	LifeMagic	S_LifeMagic
34	WarMagic	S_WarMagic
35	Leadership	S_Leadership
36	Loyalty	S_Loyalty
37	Fletching	S_Fletching
38	Alchemy	S_Alchemy
39	Cooking	S_Cooking
40	Salvaging	S_Salvaging
41	TwoHandedCombat	S_TwoHandedCombat
42	Gearcraft (deprecated)	S_Gearcraft
43	VoidMagic	S_VoidMagic
44	HeavyWeapons	S_HeavyWeapons

(continues on next page)

Table 15: (continued)

<u>Code</u>	<u>Skill (on menu)</u>	<u>myutilootor</u>
45	LightWeapons	S_LightWeapons
46	FinesseWeapons	S_FinesseWeapons
47	MissileWeapons	S_MissileWeapons
48	Shield	S_Shield
49	DualWield	S_DualWield
50	Recklessness	S_Recklessness
51	SneakAttack	S_SneakAttack
52	DirtyFighting	S_DirtyFighting
54	Summoning	S_Summoning

Table 16: Code-ordered ObjectClass codes.
(See [Table 4](#) for menu-ordering of these entries.)

<u>Code</u>	<u>ObjectClass (on menu)</u>	<u>myutilootor</u>
0	Unknown	C_Unknown
1	MeleeWeapon	C_MeleeWeapon
2	Armor	C_Armor
3	Clothing	C_Clothing
4	Jewelry	C_Jewelry
5	Monster	C_Monster
6	Food	C_Food
7	Money	C_Money
8	Misc	C_Misc
9	MissileWeapon	C_MissileWeapon
10	Container	C_Container
11	Gem	C_Gem
12	SpellComponent	C_SpellComponent
13	Key	C_Key
14	Portal	C_Portal
15	TradeNote	C_TradeNote
16	ManaStone	C_ManaStone
17	Plant	C_Plant
18	BaseCooking	C_BaseCooking
19	BaseAlchemy	C_BaseAlchemy
20	BaseFletching	C_BaseFletching
21	CraftedCooking	C_CraftedCooking
22	CraftedAlchemy	C_CraftedAlchemy
23	CraftedFletching	C_CraftedFletching
24	Player	C_Player
25	Vendor	C_Vendor
26	Door	C_Door
27	Corpse	C_Corpse
28	Lifestone	C_Lifestone
29	HealingKit	C_HealingKit
30	Lockpick	C_Lockpick
31	WandStaffOrb	C_WandStaffOrb
32	Bundle	C_Bundle
33	Book	C_Book
34	Journal	C_Journal
35	Sign	C_Sign
36	Housing	C_Housing
37	Npc	C_Npc
38	Foci	C_Foci
39	Salvage	C_Salvage

(continues on next page)

Table 16: (continued)

<u>Code</u>	<u>ObjectClass (on menu)</u>	<u>myutilootor</u>
40	Ust	C_Ust
41	Services	C_Services
42	Scroll	C_Scroll

Table 17: Code-ordered String Value Key field codes.
 (See [Table 5](#) for menu-ordering of these entries.)

<u>Code</u>	<u>StringValueKey (on menu)</u>	<u>myutilootor</u>
1	Name	M_Name
5	Title	M_Title
7	Inscription	M_Inscription
8	InscribedBy	M_InscribedBy
10	FellowshipName	M_FellowshipName
14	UsageInstructions	M_UsageInstructions
15	SimpleDescription	M_SimpleDescription
16	FullDescription	M_FullDescription
21	MonarchName	M_MonarchName
25	OnlyActivatedBy	M_OnlyActivatedBy
35	Patron	M_Patron
38	PortalDestination	M_PortalDestination
39	LastTinkeredBy	M_LastTinkeredBy
40	ImbuedBy	M_ImbuedBy
43	DateBorn	M_DateBorn
184549376	SecondaryName	M_SecondaryName

Table 18: Code-ordered SalvageType codes.
 (1 through 77, omitting 3, 9, 56, 65, and 72)
 (See [Table 10](#) for menu-ordering of these entries.)

Code	SalvageType (on menu)	myutilootor
1	Ceramic	V_Ceramic
2	Porcelain	V_Porcelain
—	N/A	N/A
4	Linen	V_Linen
5	Satin	V_Satin
6	Silk	V_Silk
7	Velvet	V_Velvet
8	Wool	V_Wool
—	N/A	N/A
10	Agate	V_Agate
11	Amber	V_Amber
12	Amethyst	V_Amethyst
13	Aquamarine	V_Aquamarine
14	Azurite	V_Azurite
15	Black Garnet	V_BlackGarnet
16	Black Opal	V_BlackOpal
17	Bloodstone	V_Bloodstone
18	Carnelian	V_Carnelian
19	Citrine	V_Citrine
20	Diamond	V_Diamond
21	Emerald	V_Emerald
22	Fire Opal	V_FireOpal
23	Green Garnet	V_GreenGarnet
24	Green Jade	V_GreenJade
25	Hematite	V_Hematite
26	Imperial Topaz	V_ImperialTopaz
27	Jet	V_Jet
28	Lapis Lazuli	V_LapisLazuli
29	Lavender Jade	V_LavenderJade
30	Malachite	V_Malachite
31	Moonstone	V_Moonstone
32	Onyx	V_Onyx
33	Opal	V_Opal
34	Peridot	V_Peridot
35	Red Garnet	V_RedGarnet
36	Red Jade	V_RedJade
37	Rose Quartz	V_RoseQuartz
38	Ruby	V_Ruby
39	Sapphire	V_Sapphire

(continues on next page)

Table 18: (continued)

<u>Code</u>	<u>SalvageType (on menu)</u>	<u>myutilootor</u>
40	Smokey Quartz	V_SmokeyQuartz
41	Sunstone	V_Sunstone
42	Tiger Eye	V_TigerEye
43	Tourmaline	V_Tourmaline
44	Turquoise	V_Turquoise
45	White Jade	V_WhiteJade
46	White Quartz	V_WhiteQuartz
47	White Sapphire	V_WhiteSapphire
48	Yellow Garnet	V_YellowGarnet
49	Yellow Topaz	V_YellowTopaz
50	Zircon	V_Zircon
51	Ivory	V_Ivory
52	Leather	V_Leather
53	Armoredillo Hide	V_ArmoredilloHide
54	Gromnie Hide	V_GromnieHide
55	Reed Shark Hide	V_ReedSharkHide
—	N/A	N/A
57	Brass	V_Brass
58	Bronze	V_Bronze
59	Copper	V_Copper
60	Gold	V_Gold
61	Iron	V_Iron
62	Pyreal	V_Pyreal
63	Silver	V_Silver
64	Steel	V_Steel
—	N/A	N/A
66	Alabaster	V_Alabaster
67	Granite	V_Granite
68	Marble	V_Marble
69	Obsidian	V_Obsidian
70	Sandstone	V_Sandstone
71	Serpentine	V_Serpentine
—	N/A	N/A
73	Ebony	V_Ebony
74	Mahogany	V_Mahogany
75	Oak	V_Oak
76	Pine	V_Pine
77	Teak	V_Teak