

# Conversation Functions & Routines:

## Ultima VII

Created 7/10/91  
By John Watson

### Functions:

#### **~PartyNPCs ();**

Gets the list of NPC numbers of the characters in the party.

Example: Party = ~PartyNPCs (); /\* Variable Party now equals the list of the party members\*/

#### **~NPCToString (NPC #);**

Converts an NPC number to the string it represents.

Example: Dupre = ~NPCToString (\_Dupre); /\*Variable Dupre now equals the string "Dupre"\*/

#### **~MoveItem (Source, Quantity, Item, Quality, Destination);**

Moves, creates, deletes, and changes the quality of items.

Source: NPC #, \_Party (or -1), \_Storage (or -2), \_Create (or -3)

Quantity: Any number (within reason)

Item: Item number (underscore constants yet to be defined)

Quality: 0 to 255 (or -1 for don't change current quality)

Destination: NPC #, \_Party (or -1), \_Storage (or -2), \_Destroy (or -3)

Example: ~MoveItem (\_Dupre, 1, \_Sword, -1, \_Avatar);

*Needs to be able  
to perform operations on items  
not in ~~the~~ inventory  
i.e. source can be ~~the~~ item: weapon*

#### **~GetStat (NPC #, Statistic);**

Statistic: \_Str, \_Dex, \_Int, \_Hits, \_MaxHits, \_Combat, \_Magic, \_MaxMagic, \_TrainingPoints

Checks an NPC and returns the value of the statistic passed, or 0 if failed.

Example: loloMax = ~GetStat (\_lolo, \_MaxHits); /\*loloMax equals lolo's maximum hits\*/

loloHits = ~GetStat (\_lolo, \_Hits); /\*loloHits equals lolo's current hits\*/

if loloMax < loloHits { /\*If lolo's maximum hits exceeds his current hits, he needs  
healing\*/

#### **~ChangeStat (NPC #, Statistic, + or - 1 to 255);**

Allows an NPC's statistics to be changed, or returns false if unable to.

Example: ~ChangeStat (\_lolo, 10); /\*Attempts to give 10 hit points to lolo\*/

**Note about ~GetStat and ~ChangeStat:** If you wished to find out the difference between lolo's maximum (loloMax) and current (loloHits) hitpoints, then heal lolo back up to full, you would type:

Heal = loloMax - loloHits; /\*If loloMax were 15 and loloHits were 8, then Heal would be 7\*/

~ChangeStat (\_lolo, Heal); /\*This will add Heal (7) to lolo's current hit points, taking him to his  
maximum allowed hit points\*/

#### **~IsNear (Item or NPC #);**

Finds out if a character or item is on screen.

Example: if ~IsNear (\_lolo); /\*Checks to see if lolo is on screen\*/

#### **~NPCStatus (NPC #);**

Checks a character for its status.

Example: Condition = ~NPCStatus (\_Shamino); /\*Condition equals the status(es) Shamino has\*/

(This should be able to return a list of statuses if there are more than one.)

**~InInv (Subject, Quantity, Item, Quality);**

**Subject:** NPC #, \_Party (or -1)

**Quantity:** Any number (within reason)

**Item:** Item number (underscore constants yet to be defined)

**Quality:** 0 to 255 (or -1 to ignore quality)

Finds out if an item is anywhere in a character or the Party's inventory(ies).

Example: if ~InInv (\_Bart, 2, \_Key, 15); /\*Checks to see if Bart has 2 keys of quality 15\*/

**~AvatarName ();**

Returns the string for the character name that the player input at the beginning of the game.

Example: Avatar = ~AvatarName (); /\*If the player chose the name Fred, Avatar now equals "Fred".\*/

["Greetings, <Avatar>! It's good to see you again, my friend."];

This prints out: "Greetings, Fred! It's good to see you again, my friend."

**~AskYesNo ();**

This will temporarily get rid of all other key words and replace them with Yes and No. One of which must be answered. (If the escape key should be ignored during this.) It returns true or false.

Example: if ~AskYesNo () {

DoThis;

}

else {

DoThat;

}

**~AskBarInput (First #, Second #, Step);**

**First #:** Lowest number of the scroll bar.

**Second #:** Highest number of the scroll bar.

**Step:** The number increment by which the mouse can move the scroll bar.

Allows input of numbers with the mouse by creating a scroll bar that displays the number selected.

Example: ["How many arrows would you like? I sell them in scores"]

Arrows = ~AskBarInput (20, 80, 20); /\*Depending on what the player sets the scroll bar to, Arrows can equal 20, 40, 60, or 80\*/

**~LordOrLady ();**

Returns the string "milord" if the Avatar is male, and "milady" if the Avatar is female.

Example: Gender = ~LordOrLady (); /\*Gender equals "milady"\*/

["Greetings, <Gender>! How can I be of assistance?"];

This prints out: "Greetings, milady. How can I be of assistance?"

(If a variable is used at the beginning of a sentence, Zack is going to have to capitalize the first letter.)

**~AvatarSex ();**

Returns the value 0 if the Avatar is male, or 1 if female.

(This is much like ~LordOrLady (); except that it doesn't return a string. It is useful, if you are just checking to see what sex the Avatar is.)

Example: if ~AvatarSex () {

[Conversation used if the Avatar is female because ~AvatarSex returned 1.];

}

else {

[Conversation used if the Avatar is male because ~AvatarSex returned 0.];

}

### **~GetTime ();**

Returns the number that represents which period of time (schedule node) the game is currently in.

Example: if ~GetTime () = 0 {

[Avatar, do you realize it's midnight. Where are your manners.];

### **~GetContents (Container, Quality, Item, Quality);**

Allows you to search the contents of a **Container** for a specific **Item**. The **Container** must have a quality but you may search for an item within it by specifying -1 for the **Item's** quality.

Example: if ~GetContents (\_Chest, 1, \_Gold, -1) {

or

if ~GetContents (\_Package, 1, \_Plans, -1) {

### **~GetQuantity (Subject, Item);**

**Subject:** NPC #, \_Party (or -1)

This function returns the number if the item specified that the **Subject** is carrying.

Example: PartyGold = ~GetQuantity (\_Party, \_Gold); /\* If the party had 500 gold coins,  
PartyGold would equal 500 \*/

### **Routines:**

#### **SetSpeaker (NPC #);**

Brings up a portrait for a character if one doesn't already exist on screen, and makes the text appear as if it belongs to this NPC. If a portrait is already up for the character, it just switches the text to the selected NPC. (Remember, only three portraits are allowed at one time: the Avatar, the person he's speaking to, and you can switch out other characters using this and CloseSpeaker (); described below.)

Example: SetSpeaker (\_Lord\_British);

#### **CloseSpeaker (NPC #);**

Removes the portrait and text, if any, of the selected NPC. (Don't use this on the currently active speaker.)

Example: CloseSpeaker (\_Lord\_British);

#### **TurnOn ("key word", "key word");**

Allows you to specify a word, or words, to be activated in the players menu of key words.

Example: TurnOn ("name", "job", "Britainnia", "AAAaarrggghhh!");

#### **TurnOff ("key word", "key word");**

Allows you to specify a word, or words, to be deactivated in the players menu of key words.

Example: TurnOff ("triscadecaphobia");

#### **SetMode (NPC #, Mode);**

This is used to set an NPC to a certain mode through conversation. You might want a character to break off conversation and attack the Avatar, or fall asleep, or even possibly die.

Example: [lolo gasps his last, then falls dead at your feet. How poignant];

SetMode (\_lolo, \_Dead);

terminate;

#### **JoinParty (NPC #);**

In order to have a character join the party, you must use this function.

Example: [Trellek sneaks into your pouch. You get a feeling of safety and comfort from him.];

JoinParty (\_Trellek);

#### **LeaveParty (NPC #);**

SetClock (ClockName);  
~GetClock (ClockName);



This does the reverse of JoinParty (). Takes an NPC out of the party.

Example: [You slap Dupre, saying, "That's the last time you goose me, you drunken sot!"];  
LeaveParty (\_Dupre);

### SetTime (Days, Hours);

This will increment the Day and Hour of the game clock in addition to actually changing the game time.

Example: [The baker finishes the bread you asked for in a couple hours then gives it to you.];  
SetTime (0, 2);

### ClearKeys ();

This routine clears all key words in the current conversation.

Example: TurnOn (<<"name", "job">>); /\* Current keys: name, job \*/  
ClearKeys (); /\* No keys \*/

### PushKeys ();

Routine that allows you to temporarily set aside keywords in order to use another set. To be used with #10, PopKeys ().

### PopKeys ();

This allows you to restore the last set of keys Pushed. This may be used to restore as many sets of keys as you've Pushed in reverse order.

Example: TurnOn (<<"name", "job">>);  
PushKeys (); /\* The keys "name" and "job" are temporarily gone \*/  
TurnOn ("La, la, la");  
PushKeys (); /\* "La, la, la" is gone, but will come Pop out before "name" and "job" \*/  
PopKeys (); /\* "La, la, la" is back \*/  
PopKeys (); /\* "name" and "job" are back, along with "La, la, la" \*/

### List of Conventions:

#Blah	Useable
\$Blah	Global Flag
_Blah	Constant
-Blah ()	Function
Blah ()	Routine
BlahBlah	Local Variable

~ Set met (~numzid) ?  
~ get met (~numzid) ?

Temp Party (N / 1000)

### Usefull AGIL Code Words:

terminate; (Breaks out of an entire useable.)

break; (Breaks out of a loop. I.e. While, Converse, Foreach, etc.)

funcit)

~ IShowFAR (\$Source, DEST >>)

~ FRAME (\$C) /

~ D ANIML; /

~ GETITEMDATA (Item, VAR'S VAR'S ANY Item data)

~ SETITEMDATA (Item, VAR'S VAR'S ANY Item data, contents, etc)

~ DIGIT (X, Y) /

~ IS ON (Item, Item) /

~ GETITEMNUMBER (SELF / INBOW)

~ GETITEM (Item)  
~ GETITEM (Item)  
~ GETITEM (Item)

Routine

SET Item (~Item, FRAME) ?

~ SET ANIM (~Item, 00-1) ?

Post on Adrenaline (

```
~JoinParty(_npc,_True/false);  
Returns false if there is a failure
```

```
check=~JoinParty(_dupre,_true);
```

Ones I have to do myself

```
~DigHere(X,Y);
```

Digs at coord returns false if was non digable area(imp in agil)

```
~IoloSays("string of outgoing smartass comments");
```

gets party, check to see if Iolo, Shamino,Dupre, sparks are in party ifso say text,  
else get last member of party have them say text close conversation last member

Routines

```
~Armour();
```

Calls IoloSays(" You really should try putting that on, it works much better that way");