

## How do I do this?

(pending revision – 16 March 2001)

### Introduction

This document is in the form Question/Objective, followed by the Runescript code to implement it. I hope you find it useful.

I also took the liberty of copying examples from a number of documents and posts. To the authors, I would like to express my thanks for permission to recompile them here, and my apologies for being so abusive of their trust, namely Dan Schnake, Adam West, and Darwin. Whenever possible, credit is given to the original author or authors.

In the process of checking this document, you may find that some written down examples by their authors are modified. This in no way detracts their logic, but I tried to validate as much as possible, by at least looking at the actual scripting produced by Creator for each statement.

### Part I – The easy stuff

*Finding if a certain type of monster is in the encounter (from Encyclopedia Arcana):*

```
ForEach CreatureA In EncounterNow  
  IfText CreatureA.Name Like Troll Then  
    DialogShow Says I see a troll!  
  EndIf  
Next
```

Comments: Notice the use of the handle CreatureA to go through all the creatures in the current, active encounter, and the use of Like to see if any of them has Troll in its name. Where DialogShow is you should put whatever you want to do if there are trolls in the encounter. If you use the code as it is, those statements will be performed once for each troll. If you want to do it just once use instead:

```
ForEach CreatureA In EncounterNow  
  IfText CreatureA.Name Like Troll Then  
    DialogShow Says I see a troll!  
    Exit Loop  
  EndIf  
Next
```

This way, you will exit the loop, after processing the first occurrence of a Troll.

*Healing all party members who are still alive (from Encyclopedia Arcana):*

```
ForEach CreatureA In Party  
  If CreatureA.Health > Pos.0 Then  
    Let CreatureA.Health = CreatureA.HealthMax  
  EndIf  
Next
```

Comments: This loop even "heals" party members that are at maximum health. A variation of this could be:

```

ForEach CreatureA In Party
  If CreatureA.Health > Pos.0 And CreatureA.Health < CreatureA.HealthMax Then
    Let CreatureA.Health = CreatureA.HealthMax
    DialogShow DM Says [Brief Box] [CreatureA] regains full health.
  EndIf
Next

```

This way, only members that require healing are handled, and the DM tells the player who has Been healed through a box that disappears by itself.

*Checking to see if the current character is holding a sword in hand (from Encyclopedia Arcana):*

```

ForEach ItemA In CreatureNow
  IfText ItemA.Name Like Sword Then
    And ItemA.IsEquipped = Global.True Then
    DialogShow Says Yep, he's holding a sword!
  EndIf
Next

```

Comments: Not really, just a compounded example to check the gender of the character and show an adequate dialog (it also illustrates in a simple way the nesting of If statements):

```

ForEach ItemA In CreatureNow
  IfText ItemA.Name Like Sword Then
    And ItemA.IsEquipped = Global.True Then
    If CreatureNow.IsMale = Global.True Then
      DialogShow Says Yep, he's holding a sword!
    Else
      DialogShow Says Yep, she's holding a sword!
    EndIf
  EndIf
Next

```

*To cure any poison with a magic wand (from Encyclopedia Arcana):*

```

ForEach TriggerA In CreatureTarget
  If TriggerA.IsPoison = Global.True Then
    Destroy TriggerA
    DialogShow Says [CreatureTarget.Name] is cured!
  EndIf
Next

```

Comments: This needs a bit of clarifying, as examples go, because it's not immediately obvious why there are triggers involved (and one of them is not even shown). The Creature.Target was poisoned before; to that effect, a trigger with poison was "attached" to Creature.Target (with a CopyTrigger statement from the original trigger that did the "attachment" of this property). To Destroy that poison property, you have to destroy the sub-trigger that Creature.Target has. This is the background.

To perform the action in the example, the character doing it has a, say, Wand of Poison Healing, which MUST have a trigger to perform its work (in fact, an On-UseOnCreature trigger), with the script of the example. Notice you do not require sub-triggers to do this, just a trigger with the Wand.

*This one uses a loop within a loop to see if any party member has the enchanted key (from Encyclopedia Arcana):*

```

ForEach CreatureA in Party
ForEach ItemA in CreatureA
  IfText ItemA.Name Like enchanted key Then
    DialogShow Says The door will open!
  EndIf
Next
Next

```

Comments: Again, this is the script of a trigger of an enchanted locked door the party is trying to open, for instance. You could expand it to cover other situations:

```

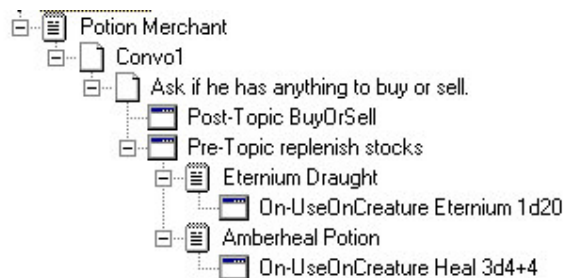
ForEach CreatureA in Party
ForEach ItemA in CreatureA
  IfText ItemA.Name Like enchanted key Then
    DialogShow Says The door will open!
  Else
    IfText ItemA.Name Like key Then
      DialogShow Says [Brief Box] [Creature.A] doesn't have the proper key!
    EndIf
  EndIf
Next
Next

```

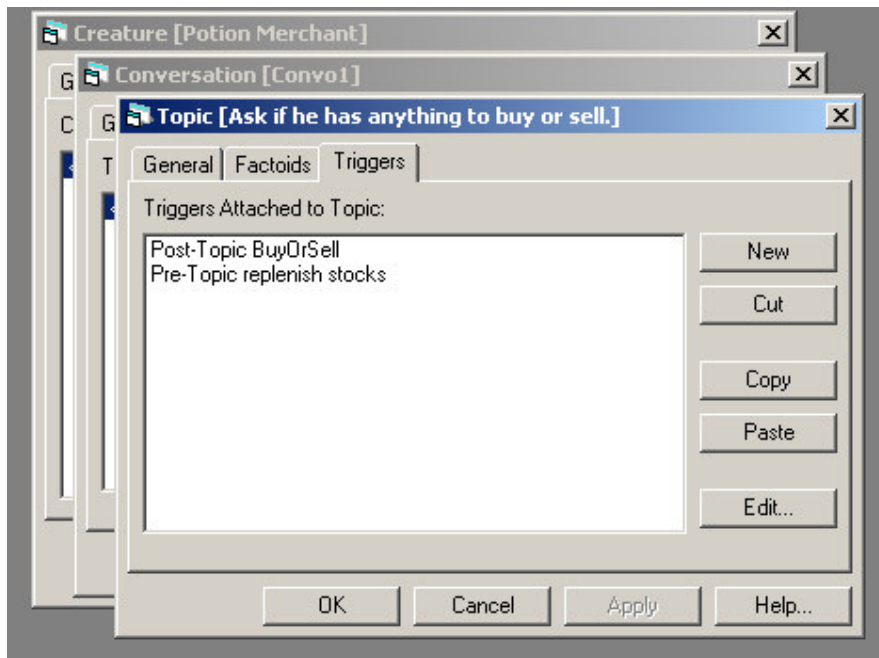
Notice that the order of the nested IfText's is important, because you want to test first for enchanted keys, as you are performing the test with Like, and not with an exact expression.

*How do I set up an encounter with a NPC to trade or buy stuff? (example extracted from The Land Beyond):*

Fair enough. Look at the image taken from the encounter with the NPC Potion Merchant:

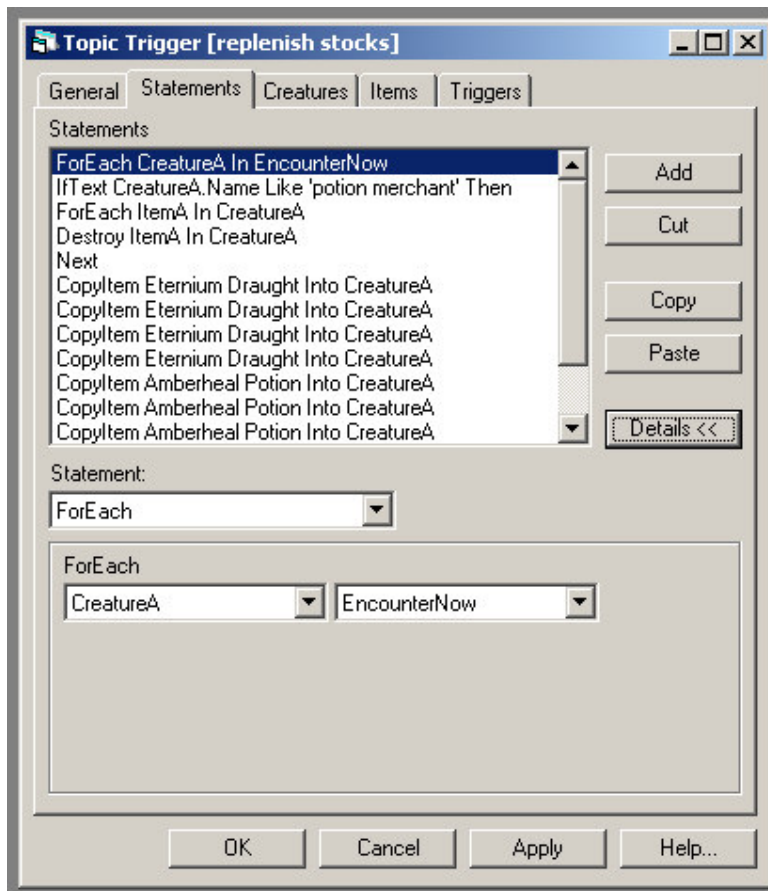


The Potion Merchant has, under the Talk tab, the default conversation, named Convo1. Convo1 has, under the General tab, a check signalling it's the default conversation, and some text that the player is shown, when approach the NPC. Under the tab named Topics, you get the expected option by default "Ask if he has anything to buy or sell". If you open up this topic, you get the details of this topic, but open the tab named Triggers:



There stands 2 triggers attached to this topic:

Post-Topic BuyOrSell fires up immediately after the topic [Ask if he has anything to buy or sell.], and the Pre-Topic replenish stocks fires immediately BEFORE the topic is initiated, so let's look at the latter first.



Here are the statements in their full glory. I've opened details, so you may see how easy it is to complete a statement. The code goes like this:

```
ForEach CreatureA In EncounterNow
  IfText CreatureA.Name Like 'potion merchant' Then
    ForEach ItemA In CreatureA
      Destroy ItemA In CreatureA
    Next
    CopyItem Eternium Draught Into CreatureA
    CopyItem Eternium Draught Into CreatureA
    CopyItem Eternium Draught Into CreatureA
    CopyItem Eternium Draught Into CreatureA
    CopyItem Amberheal Potion Into CreatureA
    CopyItem Amberheal Potion Into CreatureA
    CopyItem Amberheal Potion Into CreatureA
    CopyItem Amberheal Potion Into CreatureA
  EndIf
Next
```

Also note that in the tab called Items, you'll find 2 items attached to the trigger, obviously named Eternium Draught and Amberheal Potion. We'll look into one of them later on, but first a few comments on the script:

It loops every creature in the encounter looking for the Potion Merchant, and then loops through every item he carries and destroys them, to make sure he carries nothing; afterwards, the script copies 4 Eternium Draught items and 4 Amberheal Potions from the trigger to the Merchant. That's it! The Potion Merchant now has 8 items to sell.

I know that you could also set up 2 loops for giving the goods to the NPC, and a general example could be:

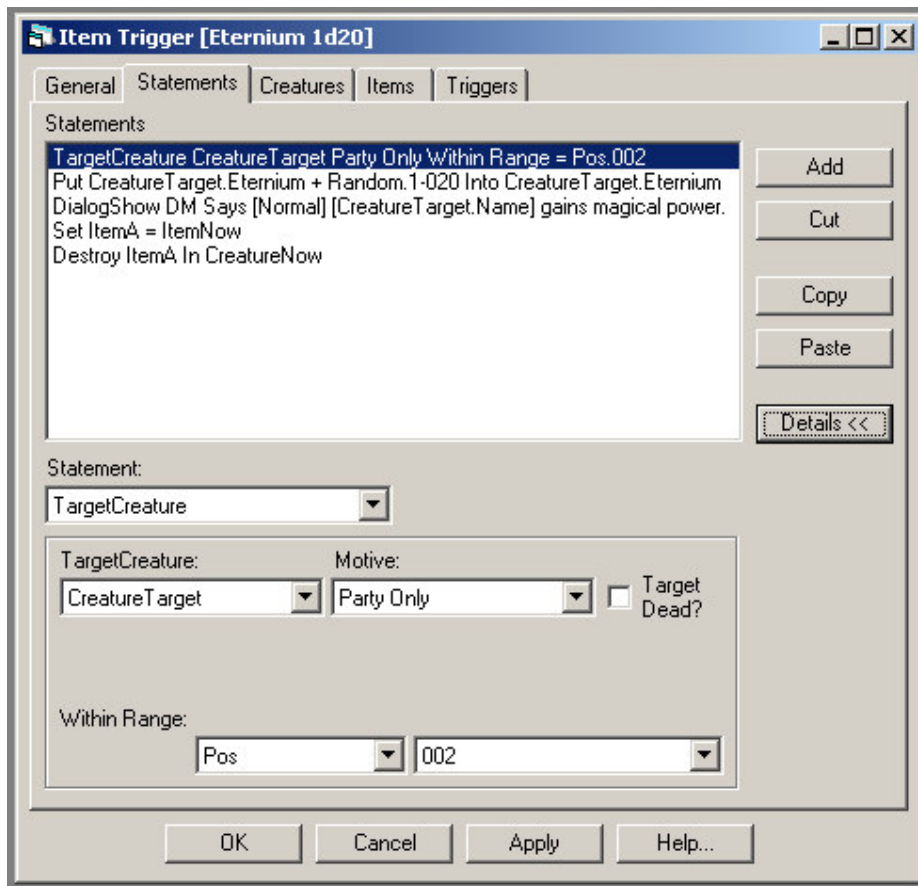
```
ForEach CreatureA In EncounterNow  
  IfText CreatureA.Name Like 'potion merchant' Then  
    ForEach ItemA In CreatureA  
      Destroy ItemA In CreatureA  
    Next  
    Let Local.ByteA = Pos.4  
    While Local.ByteA > Pos.0  
      CopyItem Eternium Draught Into CreatureA  
      Put Local.ByteA – Pos.1 Into Local.ByteA  
    Next  
    Let Local.ByteA = Pos.4  
    While Local.ByteA > Pos.0  
      CopyItem Amberheal Potion Into CreatureA  
      Put Local.ByteA – Pos.1 Into Local.ByteA  
    Next  
  EndIf  
Next
```

By changing the values in the Let statements you can actually change the number of copies of each item the Merchant has for current stock. See? That was not so hard!

But, what if one of your characters in the party was named Potion Merchant? No problem. EncounterNow does not include party members.

Finally let's look at, say the Eternium Draught (the differences to the Amberheal Potion are not in the logic of the script; look it up by yourself).

It has a On-UseOnCreature trigger attached to the item with the following script:



(this is the last image in this example).

What does it do?

It targets a party member within an adjacent "space", and increases his/her Eternium by a random number from 1 to 20, tells the party that the affected PC gained magical power, points to the Eternium object and destroys it (you drank it, now it's gone).

*Say you want a Flask of Full Eternium Recovery, a difficult and pricey item to find, due to Eternium volatile properties. How do you change the script, so that a magical character regains his/her full magical potential? It could look like this:*

```
TargetCreature CreatureTarget Party Only Within Range = Pos.002
Let CreatureTarget.Eternium = CreatureTarget.EterniumMax
If CreatureTarget.IsMale = Global.True Then
    DialogShow DM Says [Normal] [CreatureTarget.Name] fully recovers his magical
power.
Else
    DialogShow DM Says [Normal] [CreatureTarget.Name] fully recovers her magical
power.
EndIf
Set ItemA = ItemNow
Destroy ItemA In CreatureNow
```

Doesn't it look cool?

*I want to add to my party a NPC, named Tirion.*

Nothing can be simpler. First you need a trigger; secondly, the trigger must include the statement CopyCreature [Creature] Into Party. The trigger must have the creature you want to copy attached to it.

It's implementation can be something like this:

- 1) Create the Character Tirion. Make sure you tick his box for friendliness;
- 2) Create the trigger for the "exchange", say Tirion joins the Party;
- 3) Code the statement.

This will be dealt in greater depth in Part XXX

Note: Theoretically, you can have 255 party members, with an average level of 255. Of course, this is not a party, it's a cataclysm! I have not checked how adding a party member in an adventure in progress impacts the game, so beware! This looks like thin ice...

\*\*\* End of Part I \*\*\*