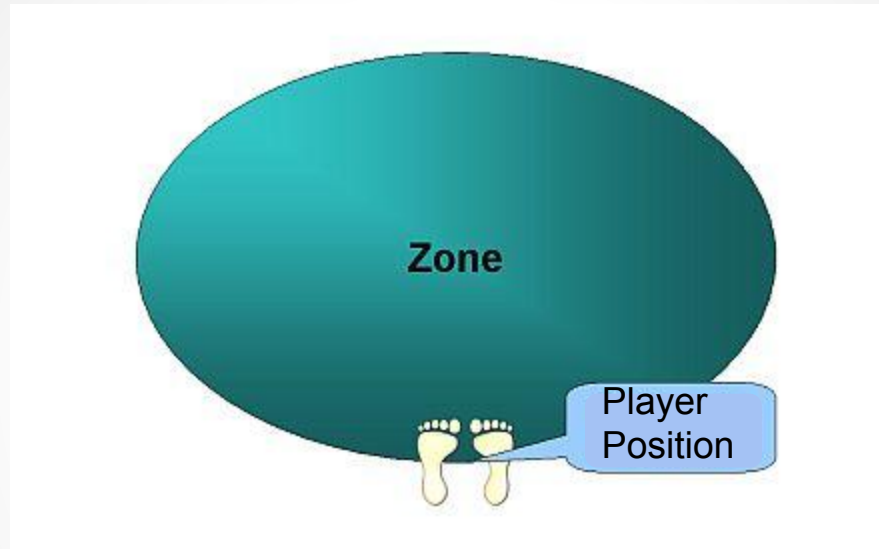


# Zone Issues

There are three **main** problems with zones:

- Zones can re-trigger because of coordinate jitter
- Items will become invisible due to coordinate jitter
- Geocachers have trouble to "enter" a zone

- zones can re-trigger because of coordinate jitter

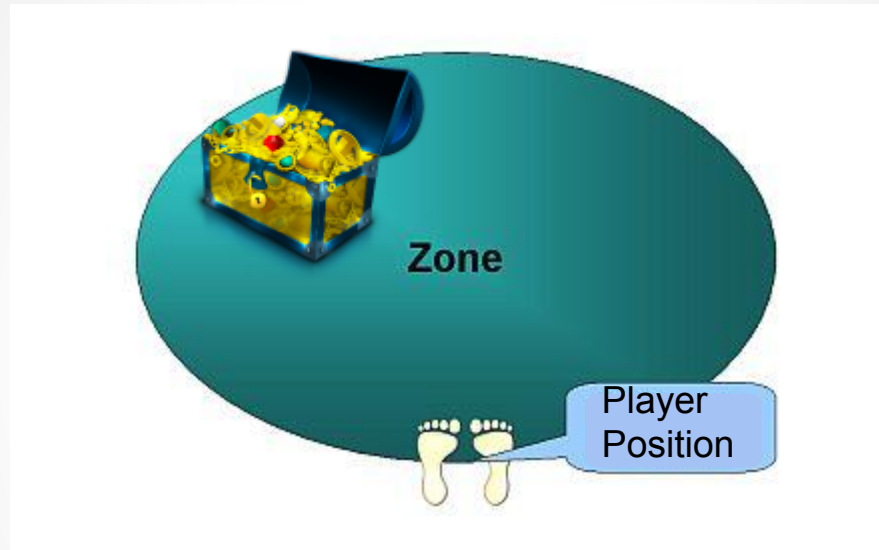


Geocachers will stop walking when the display reaches zero. Their position will typically be on the edge of the zone.

If the GPS coordinates changes slightly, the human player might toggle back and forth between just inside the zone and just outside the zone.

The OnEnter event will trigger every time that the coordinate jitter places the position inside the zone.

- items will become invisible due to coordinate jitter

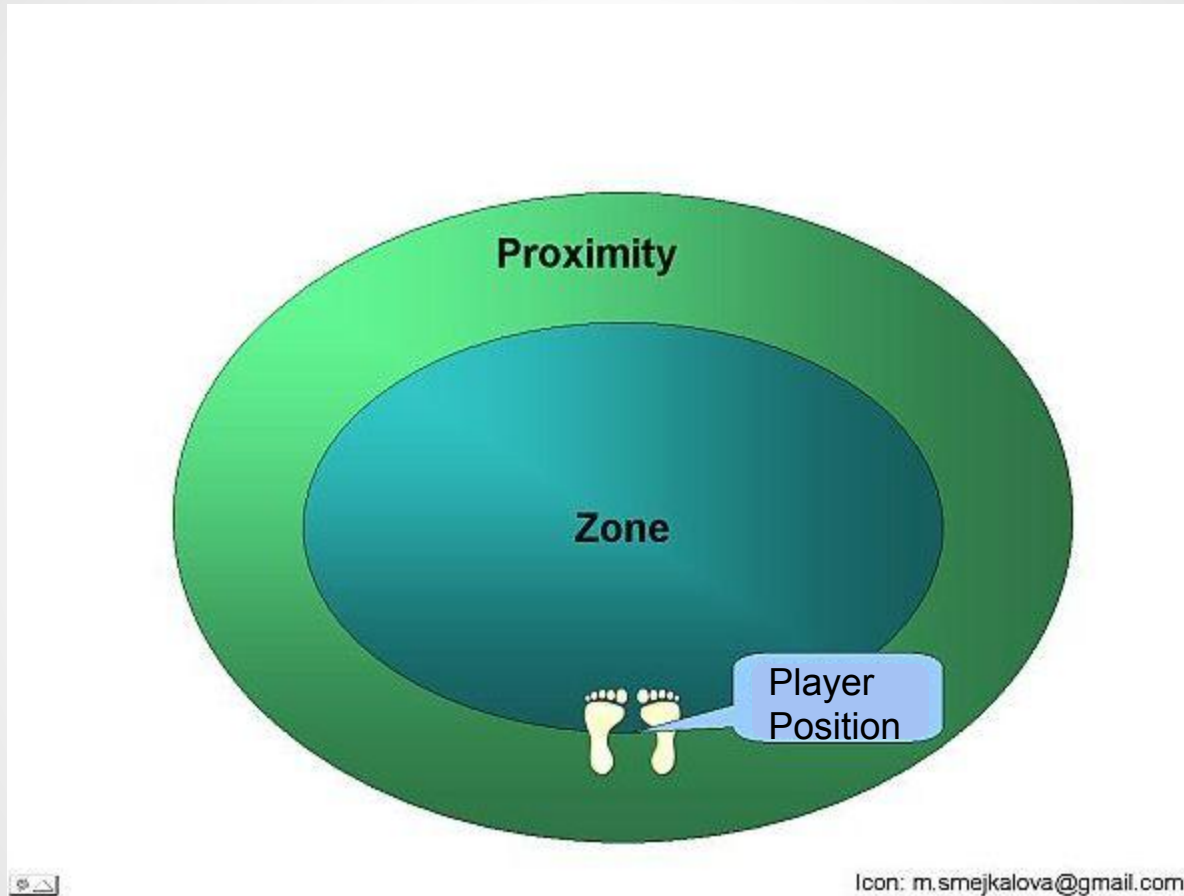


Items and characters are (by default) only visible when the human player is inside the zone (OnEnter).

As discussed before the player position might toggle between inside and outside the zone. Items will therefore become visible or invisible (usable or unusable) due to the coordinate jitter.

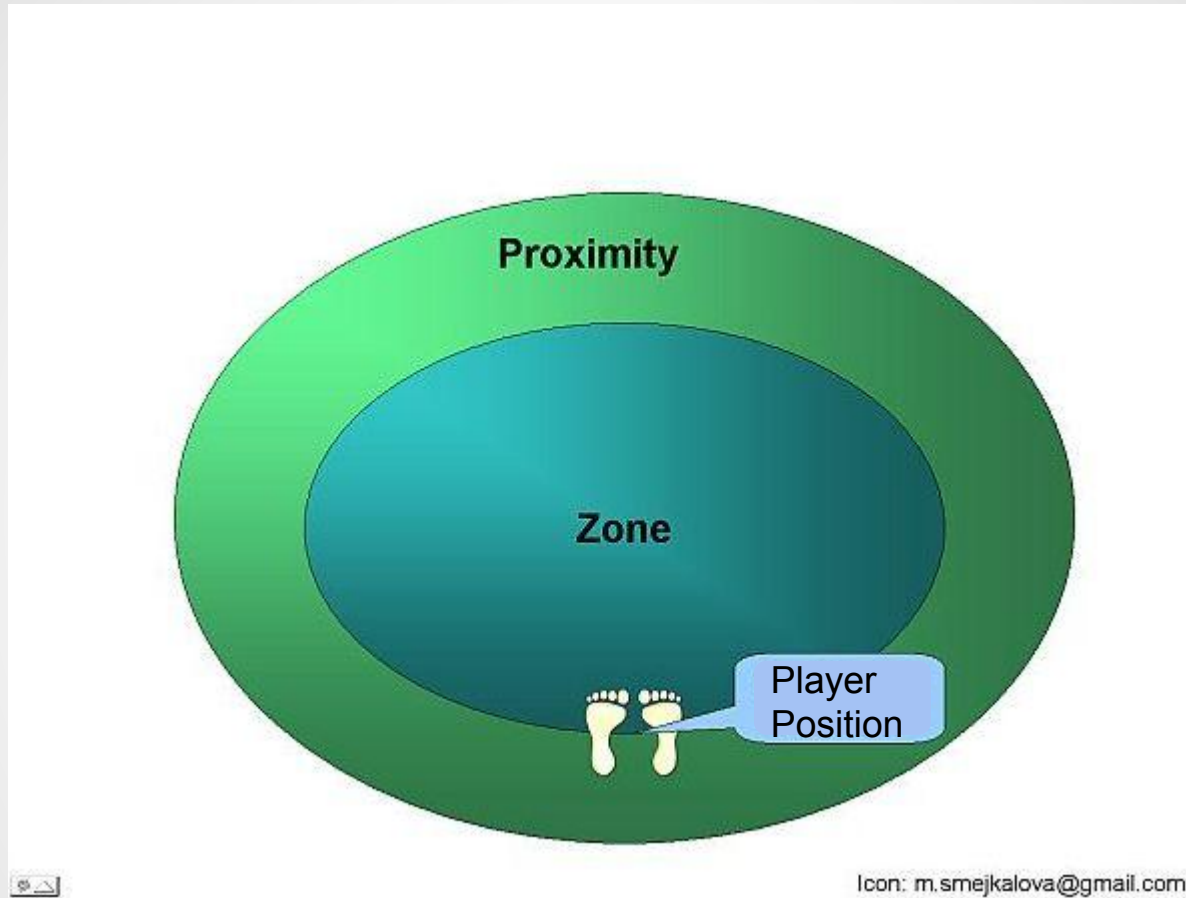
If the human player has to interact with items or characters inside the zone he might randomly be able to do so or not.

# The Comfort Zone (C) TM Trick



You can use the ShowObject property to "capture" the human player inside the proximity zone. We have called this trick Comfort Zone (C) TM. The following slides will teach you how it is done.

# The Comfort Zone (C) TM Trick



## The basic idea has two steps

- prevent the enter event of the zone from firing once the player has entered the zone. This will prevent the zone from re-triggering
- make all items visible on proximity once the player has entered the zone. This will make sure that items stay visible when the player coordinates move (jitter) the player out of the zone into the proximity layer.

# The Comfort Zone (C)™ Trick

Before you continue reading I need to find out if you are using the parameters that we use to create the Comfort Zones.

## **First parameter: ObjectVisibility (ZoneProperty: Show objects)**

Comfort Zone will temporarily change object visibility to help you recognize objects after the jitter has thrown you out of a zone. We will ask you to change your ObjectVisibility in a way that you can see the object 20 meters outside the zone once you have found the object, (which is quite natural). Object visibility will return to OnEnter after you have intentionally walked 20 meters away from the zone.

If you have never touched ObjectVisibility during programming, you should have no trouble using Comfort Zones.

If you are working with ObjectVisibility, make sure that you understand how the Comfort Zone changes ObjectVisibility (see Earwigo slides). Even if you use this value during your cartridge we think that you should have no trouble to use Comfort Zones.

# The Comfort Zone (C)™ Trick

## **Second parameter: The Proximity Range of a Zone**

Comfort Zone will set the proximity range of a zone to a value of 20 meters. This will typically have no ill effect on a cartridge.

## **Aspects to consider if you need to use your own Proximity effects together with Comfort Zone**

- If one of your zones needs to be visited more than one time during the gameplay the player needs to move away a minimum of 20 meters from the Comfort Zone before he has another chance to re-enter the zone.
- If you intend to use a ProximityRange of more than 20 meters (e.g. 100 meters) for your own effects you can happily do this with Comfort Zones but please consider one implication. If you do this to a zone that needs to be visited more than once during your game, Comfort Zone will force the player to walk the distance that you have chosen (e.g. 100 meters) away from the zone before he is allowed to trigger the Zone again.
- If you have created an extremely elaborate zone that does not cooperate with Comfort Zones, disable Comfort Zones only for this particular zone.

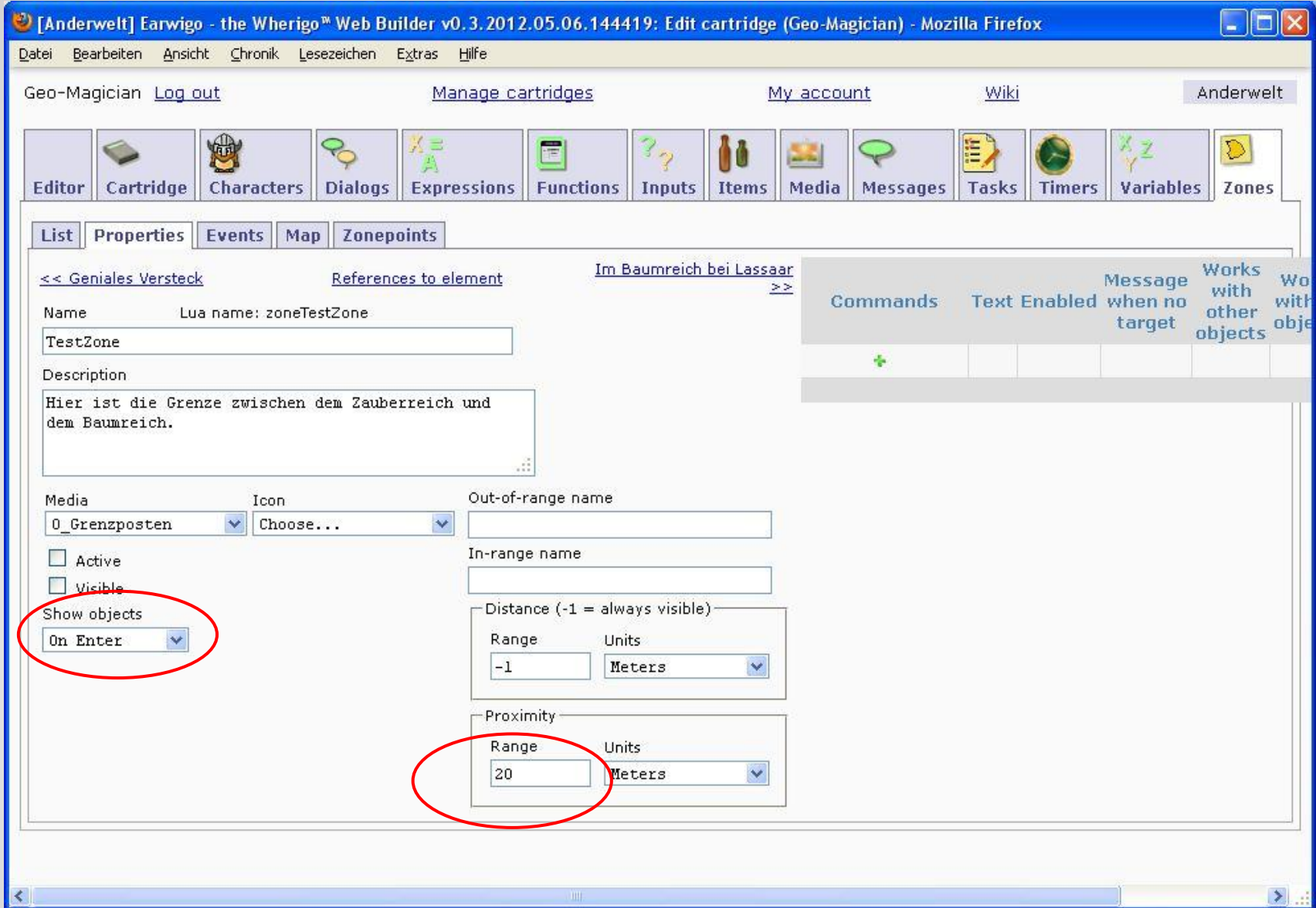
# The Comfort Zone (C)™ Trick

## Summary:

- If you are typically not using **Object Visibility** you will have no problem to use Comfort Zones
- You can use **Object Visibility** for your own projects together with Comfort Zones but you need to understand what Comfort Zones does to your Object Visibility
- If you do not plan to use **Zone Proximity** for your own effects you will have no problem to use Comfort Zones
- If you wish to use your own **Zone Proximity** effects, you will be able to use Comfort Zones, but you need to understand what Comfort Zones does to your Proximity.

The next slides will tell you how  
Comfort Zones are done in Earwigo





Set Proximity to 20 meters  
Show objects OnEnter

[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Geo-Magician [Log out](#) [Manage cartridges](#) [My account](#) [Wiki](#) [Anderwelt](#)

Editor Cartridge Characters Dialogs Expressions Functions Inputs Items Media Messages Tasks Timers Variables Zones

List Properties Events Map Zonepoints

Zone: TestZone

Name	Description
Distant (1)	When the player is within distance to the zone
Proximity (0)	When the player is within proximity to the zone
Enter (3)	When the player enters the zone
Exit (1)	When the player exits the zone
SetActive (0)	When the zone's state (active, inactive) changes

Statements

- if Has really left the TestZone
- Call function ItemVisibleOnProximity (NameOfZone=Zone:TestZone)
- Display dialog or message Dialog:RondolfsDialog
- else
- endif

Copy statements to function

All Enter events will start with the condition:  
if "Has really left the XX\_Zone"

[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Geo-Magician [Log out](#) [Manage cartridges](#) [My account](#) [Wiki](#) [Anderwelt](#)

Editor Cartridge Characters Dialogs Expressions Functions Inputs Items Media Messages Tasks Timers Variables Zones

List Properties


<< [Verlassen Genial](#) [References to element](#) [Verlassen Kissbis](#) >>

Name  Lua name:

Description

Conditions

()	+	≠	×	<u>Lua condition</u> <u>zoneTestZone.ShowObjects</u> ==
↺	+			<u>"OnEnter"</u>



"Has really left the XXZone" tests the ShowObjects property for the default value "OnEnter"

[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

[Datei](#)
[Bearbeiten](#)
[Ansicht](#)
[Chronik](#)
[Lesezeichen](#)
[Extras](#)
[Hilfe](#)

[Geo-Magician](#)
[Log out](#)
[Manage cartridges](#)
[My account](#)
[Wiki](#)
[Anderwelt](#)

[Editor](#)
[Cartridge](#)
[Characters](#)
[Dialogs](#)
[Expressions](#)
[Functions](#)
[Inputs](#)
[Items](#)
[Media](#)
[Messages](#)
[Tasks](#)
[Timers](#)
[Variables](#)
[Zones](#)

[List](#)
[Properties](#)
[Events](#)
[Map](#)
[Zonepoints](#)

Zone: TestZone

Name	Description
Distant (1)	When the player is within distance to the zone
Proximity (0)	When the player is within proximity to the zone
Enter (3)	When the player enters the zone
Exit (1)	When the player exits the zone
SetActive (0)	When the zone's state (active, inactive) changes

**Statements**

- if Has really left the TestZone
- Call function ItemVisibleOnProximity (NameOfZone=Zone:TestZone)
  - Display dialog or message Dialog:RondolfsDialog
- else
- endif

Copy statements to function

Next step in the OnEnter event is to set the ShowObject property to the proximity range:  
"Call function ItemVisibleOnProximity"



[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

Geo-Magician [Log out](#) [Manage cartridges](#) [My account](#) [Wiki](#) [Anderwelt](#)

Editor Cartridge Characters Dialogs Expressions **Functions** Inputs Items Media Messages Tasks Timers Variables Zones

**List** **Properties**

<< [ItemVisibleOnEnter](#) [References to element](#)

Name Lua name: ItemVisibleOnProximity

ItemVisibleOnProximity

Description

Sichtbarkeit der Objekte in Zone auf Proximity

Parameter name

NameOfZone

**Statements**

+ // ✖ Lua statement NameOfZone.ShowObjects = "OnProximity"

↻ +

Copy statements to function

Choose...

This is how to do it with a Function:  
ShowObjects = OnProximity

[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Geo-Magician [Log out](#) [Manage cartridges](#) [My account](#) [Wiki](#) [Anderwelt](#)

Editor Cartridge Characters Dialogs Expressions Functions Inputs Items Media Messages Tasks Timers Variables Zones

List Properties Events Map Zonepoints

Zone: TestZone

Name	Description
Distant (1)	When the player is within distance to the zone
Proximity (0)	When the player is within proximity to the zone
Enter (3)	When the player enters the zone
Exit (1)	When the player exits the zone
SetActive (0)	When the zone's state (active, inactive) changes

Statements

Call function [ItemVisibleOnEnter](#) (NameOfZone=[Zone:TestZone](#))

Copy statements to function

Copy

Use the OnDistant event to detect that the human player has really left the zone (20 meters). Now restore the ShowObjects property.

[Anderwelt] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe

Geo-Magician [Log out](#) [Manage cartridges](#) [My account](#) [Wiki](#) [Anderwelt](#)

Editor Cartridge Characters Dialogs Expressions Functions Inputs Items Media Messages Tasks Timers Variables Zones

List Properties

[References to element](#) [ItemVisibleOnProximity](#) >>

Name Lua name: ItemVisibleOnEnter

ItemVisibleOnEnter

Description

Sichtbarkeit der Objekte in Zonen zurück auf OnEnter

Parameter name

NameOfZone

Statements

+ // ✖ Lua statement [NameOfZone.ShowObjects](#) = ["OnEnter"](#)

↻ +

Copy statements to function

Choose... Copy

This is how to do it with a Function:  
ShowObjects = OnEnter

# Comfort Zone Magic

The programing effort required to achieve this benefits is substantial but it can be done.

The good news is, you can actually do this automatically. See the next slides for an introduction of how this can be done with minimal work ;-).



```

function InitZones(parCart,parZones,
parProximity)

-- Helper function for table
function IsElementInTable(parElement,parTable)
  for k,v in pairs(parTable) do
    if v == parElement then
      return true
    end
  end
  return false
end
-- Set new OnEnter an OnDistant for all zones
for k,v in pairs(parCart.AllZObjects) do
  if (parZones == nil) or (IsElementInTable(v,parZones)) then
    if Wherigo.Zone:made(v) then
      -- If parProximity not zero, then set new Proximity range
      if parProximity ~= nil then
        v.ProximityRange = Distance(parProximity, "meters")
      end
      v.RealOnEnter = v.OnEnter
      v.RealOnDistant = v.OnDistant

      function v:OnEnter()
        if self.ShowObjects == "OnEnter" then
          self.ShowObjects = "OnProximity"
          if self.RealOnEnter ~= nil then
            self:RealOnEnter()
          end
        end
      end

      function v:OnDistant()
        self.ShowObjects = "OnEnter"
        if self.RealOnDistant ~= nil then
          self:RealOnDistant()
        end
      end
    end
  end
end
end
end
end

```

# Comfort Zone Magic

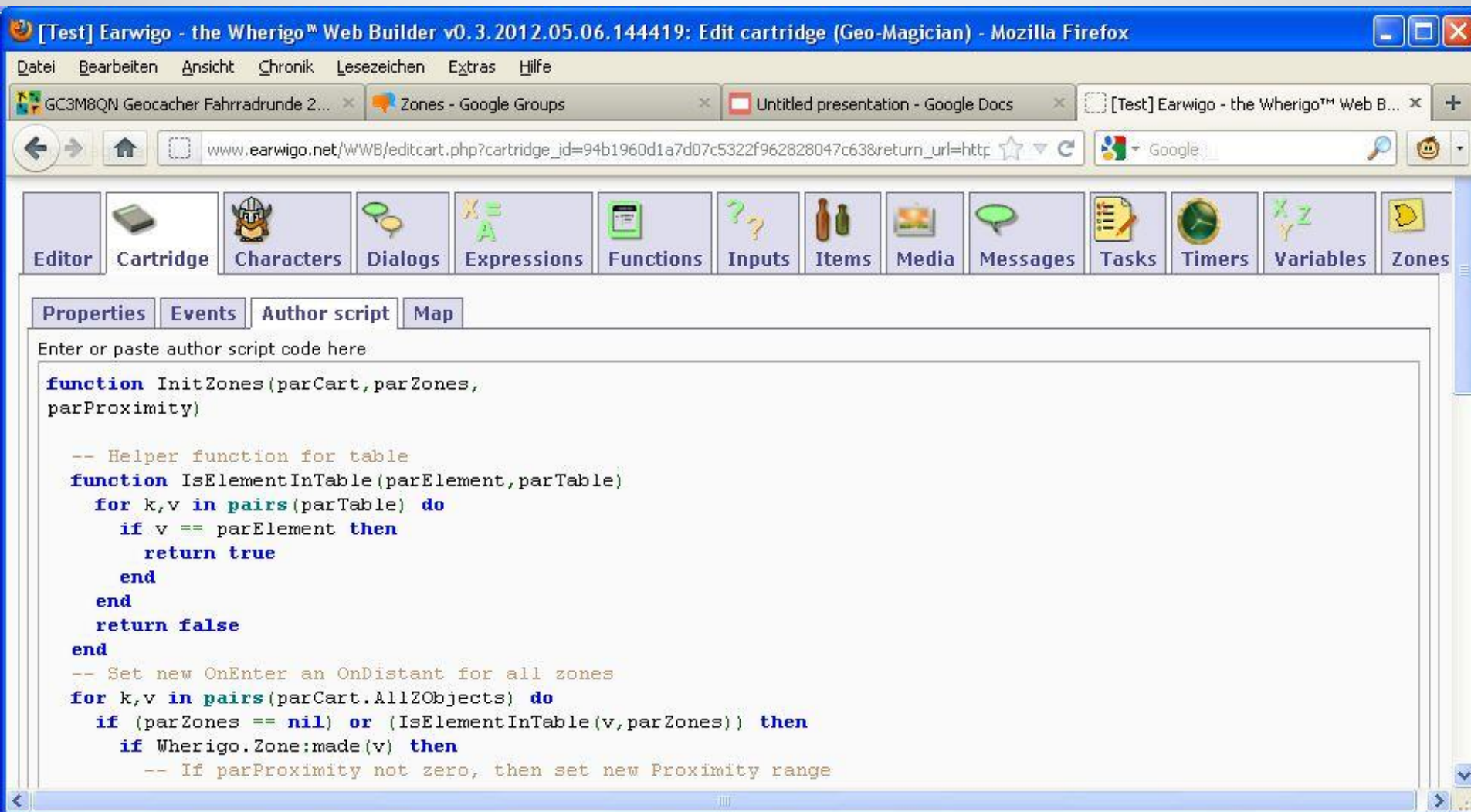
Dirk and Jan came up with the author script listed on the left side. If you run it OnStart and OnResume (Start and Resume are Events of the cartridge itself) it will automatically add the described functionality to your zones.

Include this script in your cartridge and call the function

**InitZones(self,nil,20)**

from the cartridge start event. All zones of your cartridge will automatically receive the jitter protection outlined in the previous slides.

# How to do the magic



Paste the code from the previous slide into the Author script section.

# How to do the magic

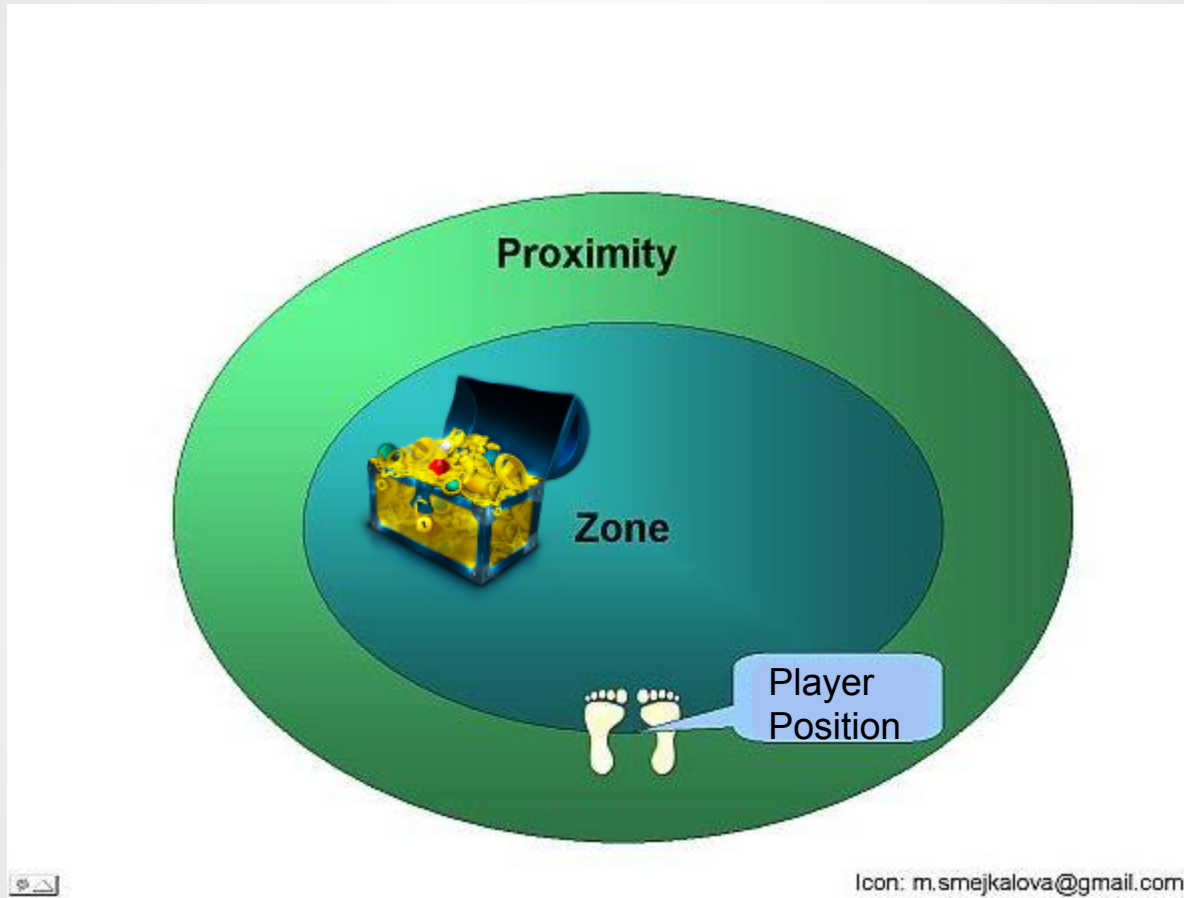
The screenshot shows the Earwigo Web Builder interface in Mozilla Firefox. The browser title is "[Test] Earwigo - the Wherigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox". The interface includes a menu bar (Datei, Bearbeiten, Ansicht, Chronik, Lesezeichen, Extras, Hilfe) and a toolbar with various icons for editing. The main content area is divided into several panels. The 'Events' panel is active, showing a table of events for the 'Cartridge: Test'.

Name	Description
Start (1)	When the cartridge is started
End (0)	When the cartridge is ended
Restore (1)	When the cartridge is resumed
Save (0)	When the cartridge is saved

Red arrows point from the 'Start' and 'Restore' events to the 'Statements' panel. The 'Statements' panel shows a list of statements. The first statement is a 'Lua statement' named 'InitZones(self,nil,20)'. A red arrow points from this statement back to the 'Start' and 'Restore' events, indicating that these events are configured to call this function.

Call the function from the Start and Restore Events.

# How to do the magic



That's it already. You are done. Your zones will behave from now on ;-)

# How does it work?

The script done by Jan and Dirk will be executed every time that your cartridge is started on a Wherigo Player unit (such as Garmin, iPhone, ...).

The script will then find all zones in your cartridge and it will automatically enhance "the zones with the required code". Programmers call this a "runtime program modification". Don't worry about this fancy terminology as long as you can believe that it works.

That's it already ... with one minor addition. Should a player save his current progress in your cartridge and restart it later on, he will not "**Start**" a cartridge, he will "**Resume**" it. That is why the Magic has to be done in the Start and in the Resume Event of the cartridge.

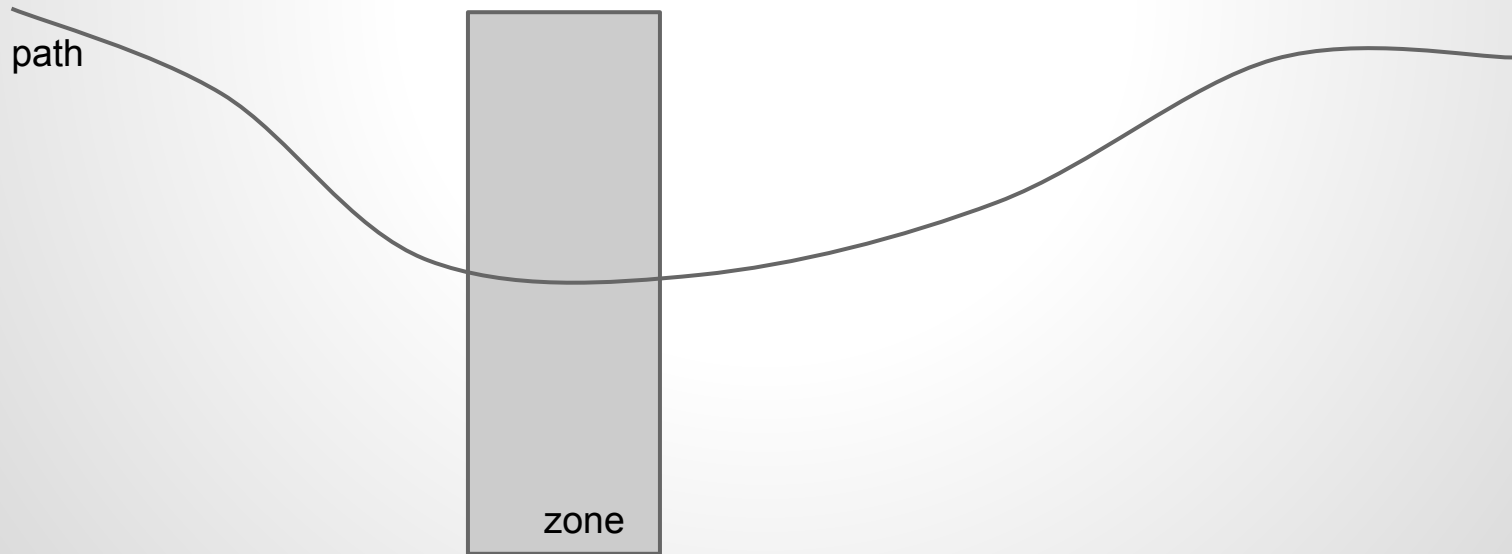
There you go. That's it!

The Comfort Zone Trick will get rid of two zone issues but there will still be the chance that geocachers might never be able to enter a zone.

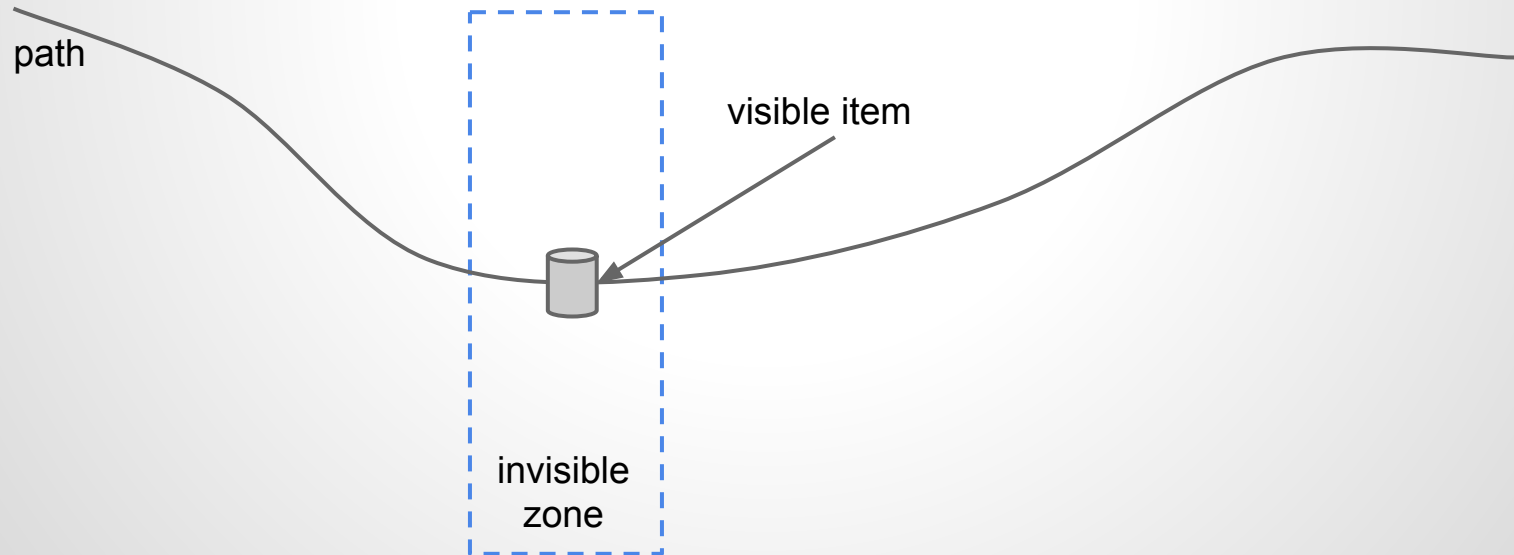
Remember: Geocachers will stop walking when the GPS reading gets close to zero.

Geocachers will stop walking when the GPS reading gets close to zero.

A first solution is to have large zones. As large as possible. Avoid small zones. Put large zones across the path to guide the geocacher into the zone.



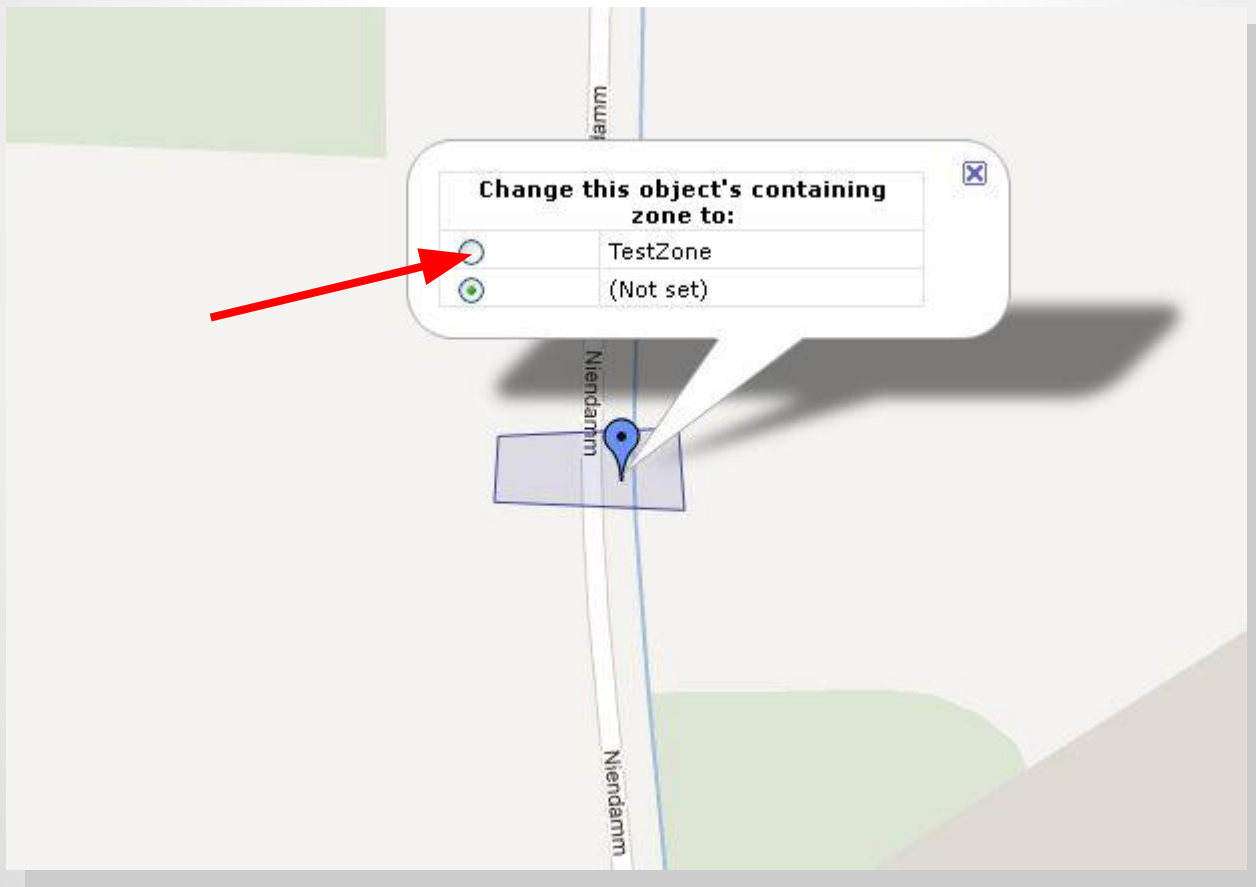
A more advanced solution is to make the zones invisible to the human player and use a visible item as the target for the human player.





Be careful to place the item inside an active zone container. It will only show up if it is placed inside the active zone.

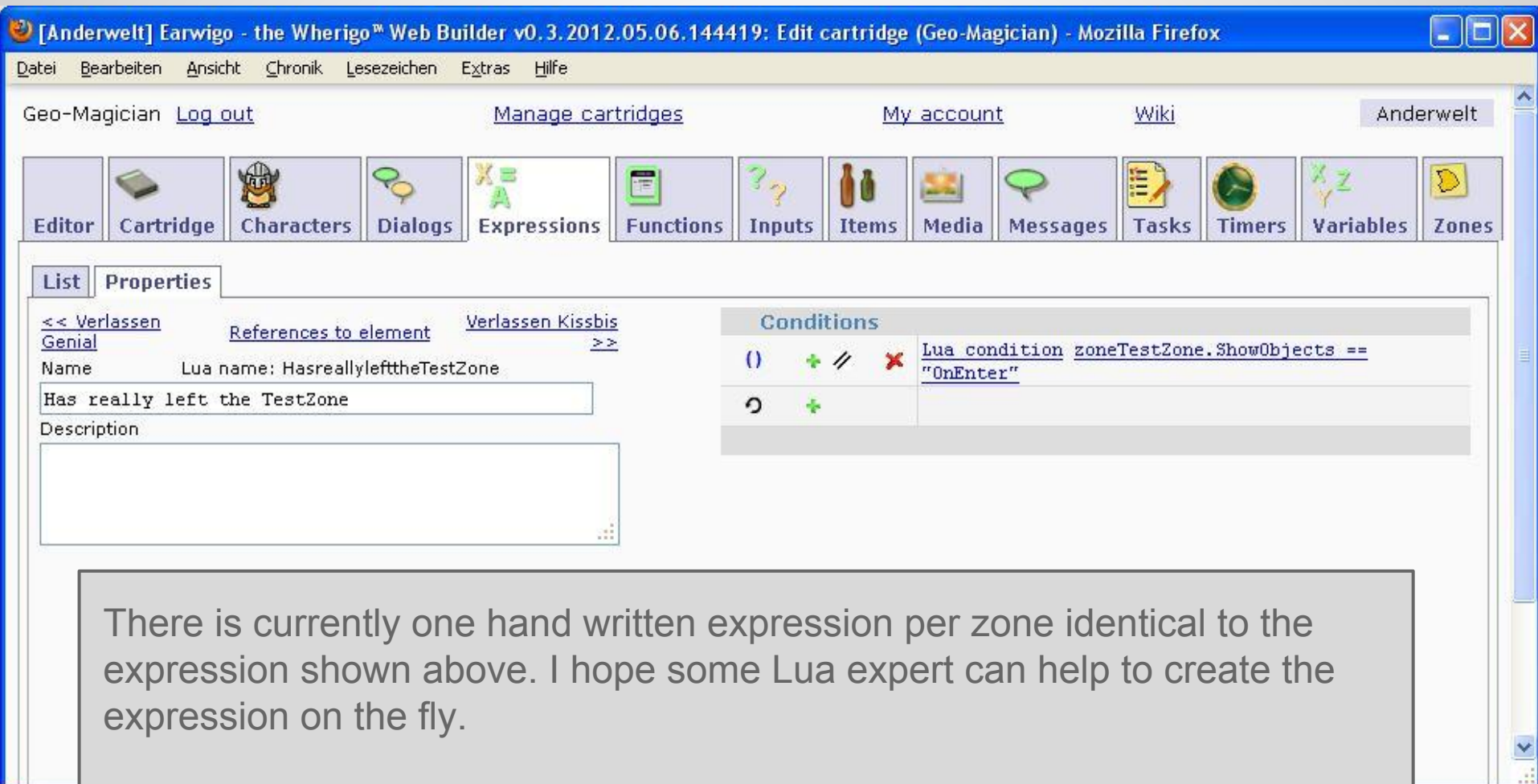
This is done by dragging the item to the zone center and selecting the respective zone (testZone in my case) in the pop-up dialog.



# **Old slide depository**

Old slides are stored after this slide for  
reference purpose

# Help wanted for the ShowObjects trick



The screenshot shows the Earwigo Web Builder interface in Mozilla Firefox. The title bar reads "[Anderwelt] Earwigo™ Web Builder v0.3.2012.05.06.144419: Edit cartridge (Geo-Magician) - Mozilla Firefox". The menu bar includes "Datei", "Bearbeiten", "Ansicht", "Chronik", "Lesezeichen", "Extras", and "Hilfe". The top navigation bar has links for "Geo-Magician", "Log out", "Manage cartridges", "My account", "Wiki", and "Anderwelt". Below this is a toolbar with icons for Editor, Cartridge, Characters, Dialogs, Expressions, Functions, Inputs, Items, Media, Messages, Tasks, Timers, Variables, and Zones. The main workspace is divided into two panes. The left pane has tabs for "List" and "Properties". The "Properties" tab is active, showing a form for an element named "Has really left the TestZone". The form includes a "Name" field with the value "Has really left the TestZone", a "Description" field, and a "Lua name" field with the value "HasreallylefttheTestZone". The right pane is titled "Conditions" and shows a table with two rows. The first row has a green plus icon, a red minus icon, and a red X icon, with the text "Lua condition zoneTestZone.ShowObjects == 'OnEnter'". The second row has a green plus icon and a red minus icon, with the text "OnEnter".

<< Verlassen  
Genial

References to element

Verlassen Kissbis  
>>

Name

Lua name: HasreallylefttheTestZone

Has really left the TestZone

Description

Conditions

()	+		×	Lua condition zoneTestZone.ShowObjects == "OnEnter"
↺	+			

There is currently one hand written expression per zone identical to the expression shown above. I hope some Lua expert can help to create the expression on the fly.

A code line similar to  
**IF** Has\_realy\_left\_the **zone: TestZone**  
would be awesome.  
(And yes, this will not work in Lua but what does? ;-)