

The Call Routing Guy

A blog by Tom Wellige in General

Followers

1



#17: Be more flexible on dates! (Part 2 - Lua)



Entry posted by Tom Wellige in Lua June 11, 2022
5,482 views

[Share](#)

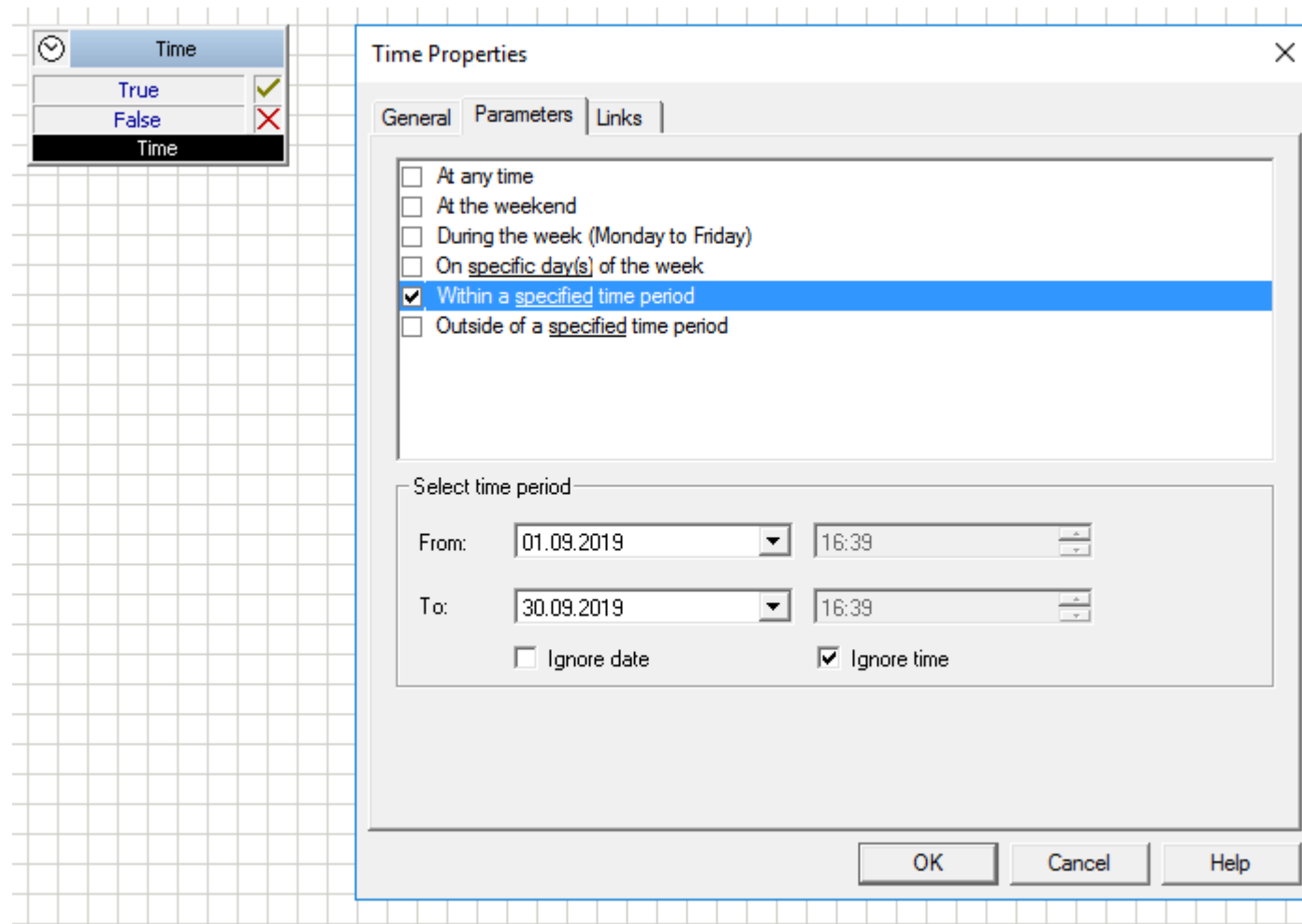
Followers

1

Lua

→ VBScript

Some time ago I wrote about checking for being in (or out) of a time period more flexible. To recap, the GSE **Time** block only offers to enter hard coded dates and times, but it doesn't offer the usage of variables.



In that previous article I showed how easy it VBScripts makes to take the date/time calculation in the own hands by using **CDate()** to convert a string into a date and afterwards use **DateDiff()** for the calculation.

The beauty of the CDate() function is, that it handles all the different date formats like **dd.mm.yyyy**, **mm/dd/yyyy**, **dd/mm/yyyy**, a.s.o. for us. You just pass a written date like "24.12.2022" into the function, which then checks the regional settings of your Windows machine to figure the correct

date/time format to identify the day, month and year correctly.

Of course Lua also has some nice date and time functionality, but it completely leaves the format trouble on the users side. To get a date value where we can calculate with there is an **os.time()** function which takes the date not as string but as a table, and we have to fill this table with all values like day, month, year, a.s.o.

This might sound more complicated as it is:

```
local vChristmas = os.time{day=24, month=12, year=2022}
```

but it becomes obvious that we can't use date strings like "24.12.2022" anymore. Unless we handle the conversion from string to table (incl. taking care of the date format) ourselves:

```
local sChristmas = "24.12.2022"

local nDay      = tonumber(StringLeft(sChristmas, 2))
local nMonth    = tonumber(StringMid(sChristmas, 4, 2))
local nYear     = tonumber(StringRight(sChristmas, 4))

local vChristmas = os.time{day=nDay, month=nMonth, year=nYear}
```

That's a lot of code just to be able to use a date string like "24.12.2022" don't you agree?

In the end I want to have a call routing which will be handled exactly like the one in the original article, where I can set a start and end date of my vacation as a GSE rule parameter, as string. But I don't want to do all this conversion stuff by myself.

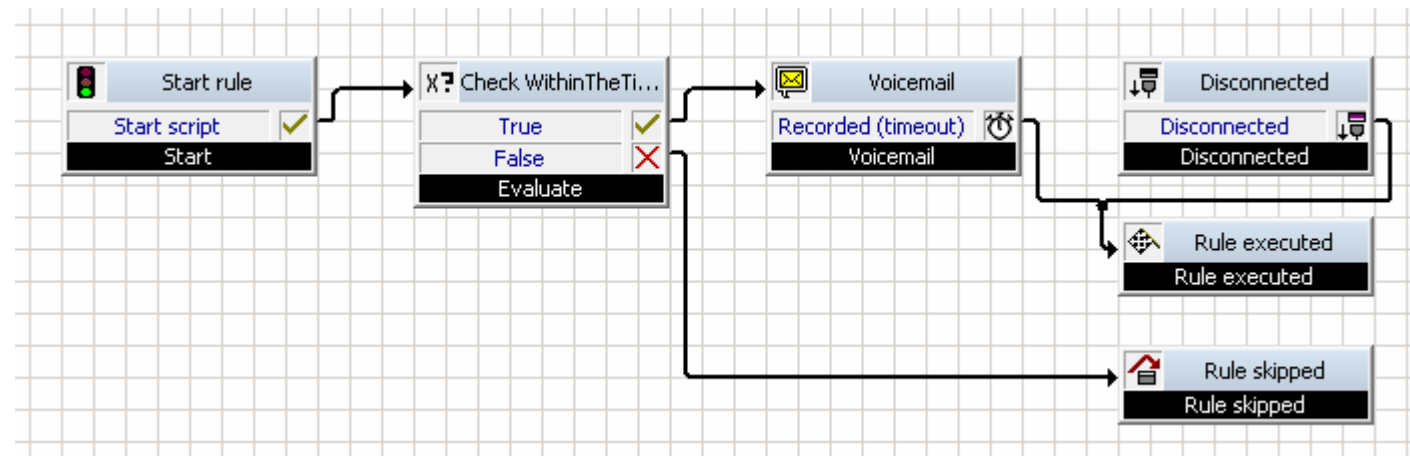
Fortunately there is a solution, and a really easy one in fact. Behind every GSE block there is some either VBScript or Lua code which handles the action or the selected condition. Behind the **Within a specified time period** condition of the **Time** block there is a function called

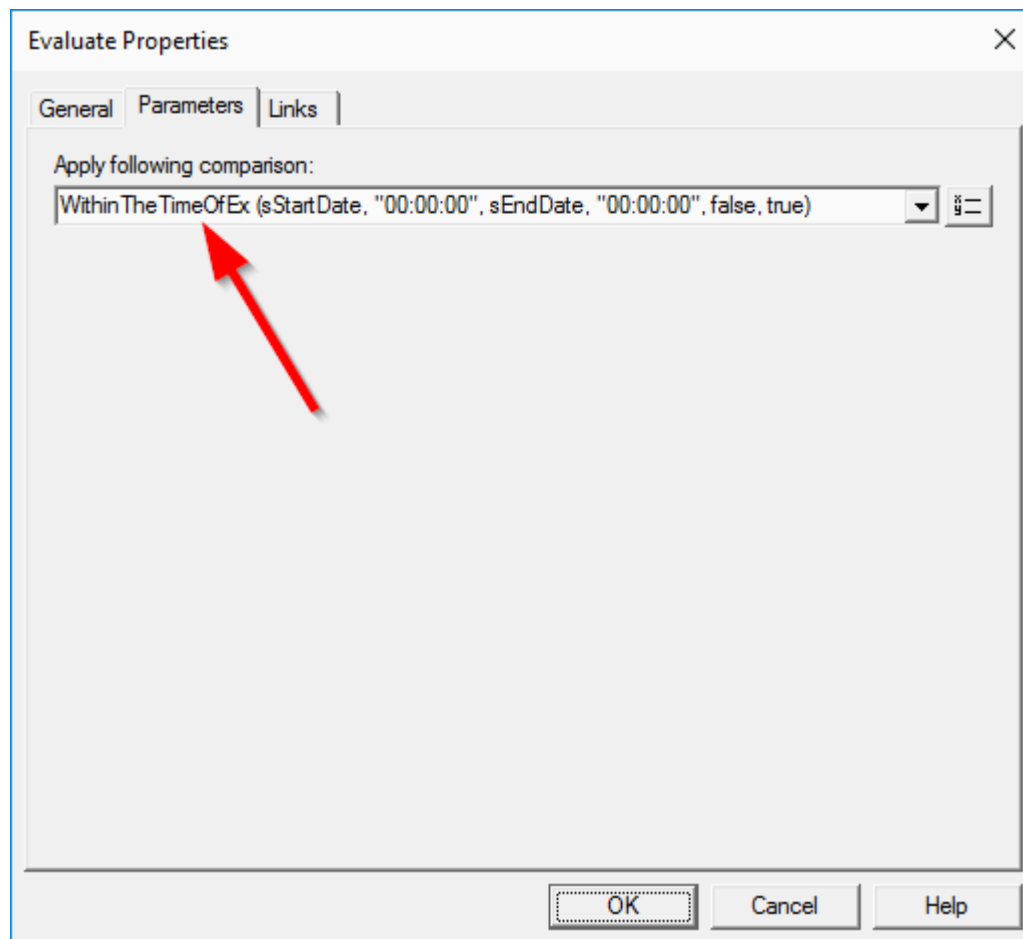
```
function WithinTheTimeOfEx(szBeginDate, szBeginTime, szEndDate, szEndTime, bIgnoreDate, bIgnoreTime)
```

It takes all the parameters from the time period configuration window (as string), does all needed conversion and afterwards also the needed calculation using **os.datediff()**.

Only thing we have to be aware of when using this function is that it uses a hard coded date format "**dd.mm.yyyy**" and time format "**HH:MM:SS**". By the way, you might have noticed from the [GSE build-in functions](#) [CurDate\(\)](#), [CurDateTime\(\)](#), [CurTime\(\)](#), and the [Server Script API](#) function [PBXUser.Now\(\)](#), that they also use hard coded date/time formats. You now have an idea why that is.

So, coming back to the call routing:





With the rule parameters

Properties ✕

General Parameters

Parameters list: + ⌂ ✕ ↑ ↓

Parameter Name	Variable Name	Default Value
Start Date	sStartDate	01.01.2000
End Date	sEndDate	01.01.2000

OK Cancel Help

Properties

General Parameters

Project name: Vacation

Script type: Rule

Author:

Company:

Build number: 1

Description:

Start Date: %sStartDate%
End Date: %sEndDate%

Create description

Block appearance

Block width (64..268): 140

Serial Numbers...

OK Cancel Help

You can download this script from here:



Vacation_Lua.rse

2.3 kB · 91 downloads

Enjoy!

PS: Don't miss to take a look into the [ECR - Useful Link Collection](#)



Previous entry

#16: Code, code, code

Next entry

#10: I can't hide anymore!



0 Comments

There are no comments to display.

Create an account or sign in to comment

You need to be a member in order to leave a comment

Create an account

Sign up for a new account in our community. It's easy!

Register a new account

Sign in

Already have an account? Sign in here.

Sign In Now

Theme ▼

Copyright (c) 2007-2025 by Tom Wellige

Powered by Invision Community