

## The Call Routing Guy

A blog by Tom Wellige in General

Followers

1



## #21: The world isn't black & white, or is it?



Entry posted by Tom Wellige in VBScript September 30, 2024  
1,326 views

[Share](#)

Followers

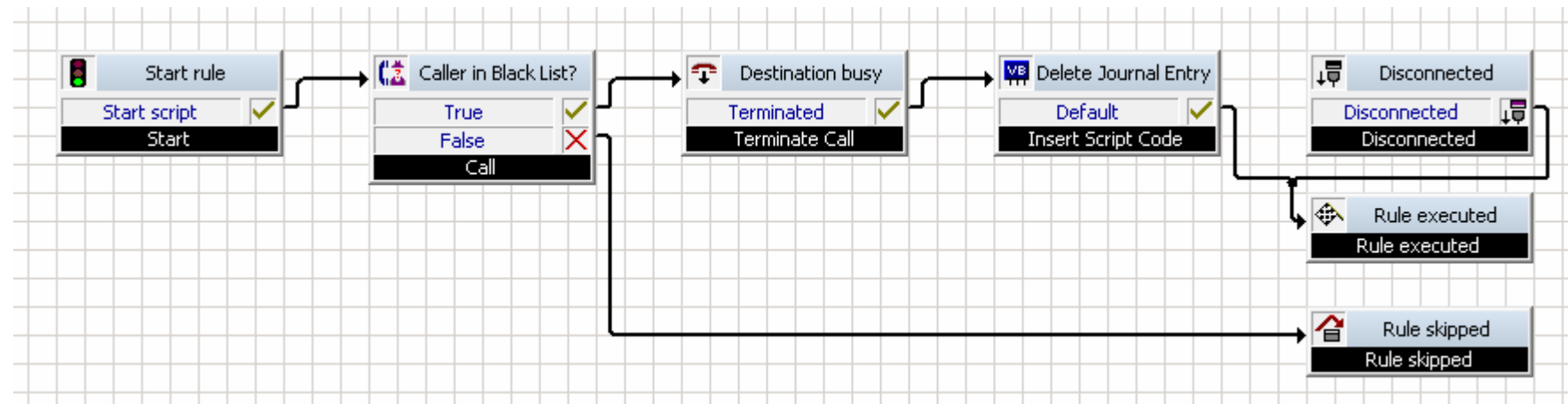
0

Today I would like to talk a little bit about **black listing** and **white listing** of calls into the SwyxWare.

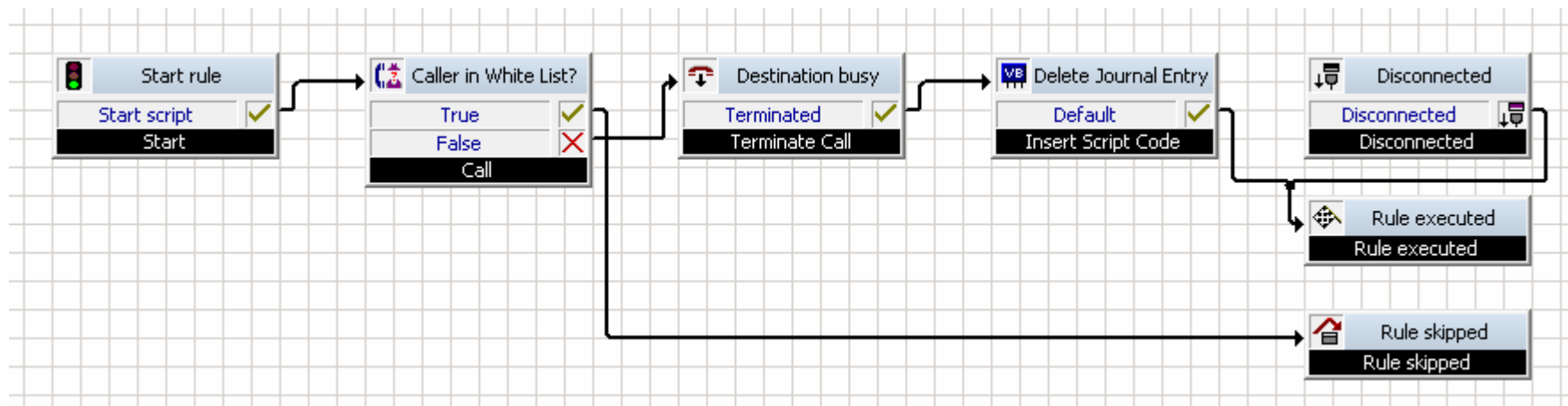
Black listing means, that if a caller number is on the so called "black list" the call will be **rejected**. All other calls will be **connected**.  
White listing means, that if a caller number is on the so called "white list" the call will be **connected**. All other calls will be **rejected**.

Both attempts can easily be implemented within the SwyxWare call routing:

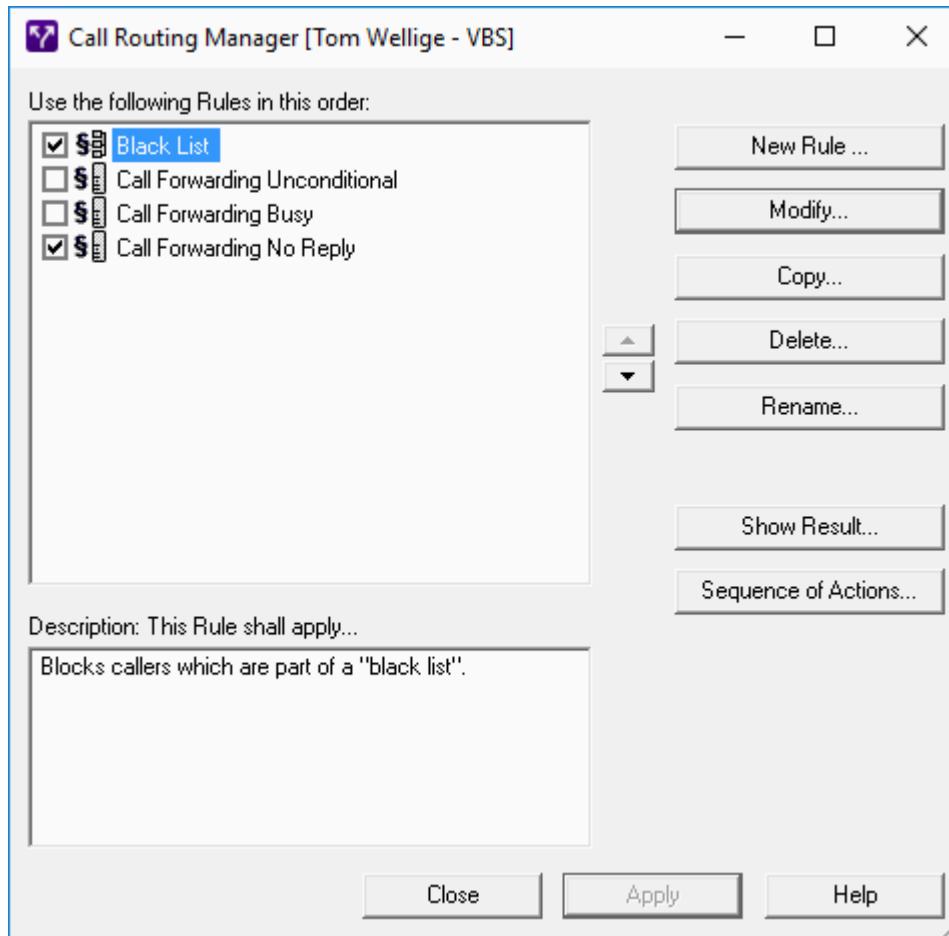
### Simple Black Listing



### Simple White Listing



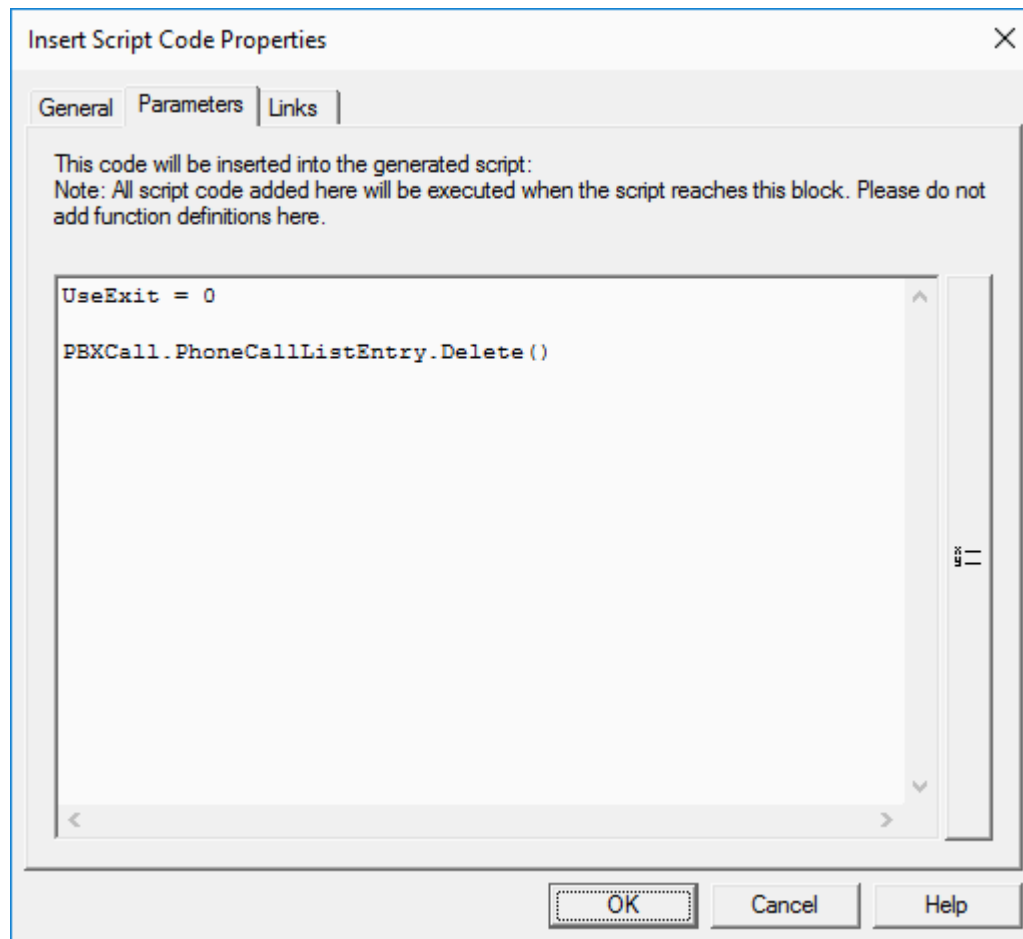
The final call routing rule should be placed **on top** of this list of rules within the users' or groups' **Call Routing Manager**.



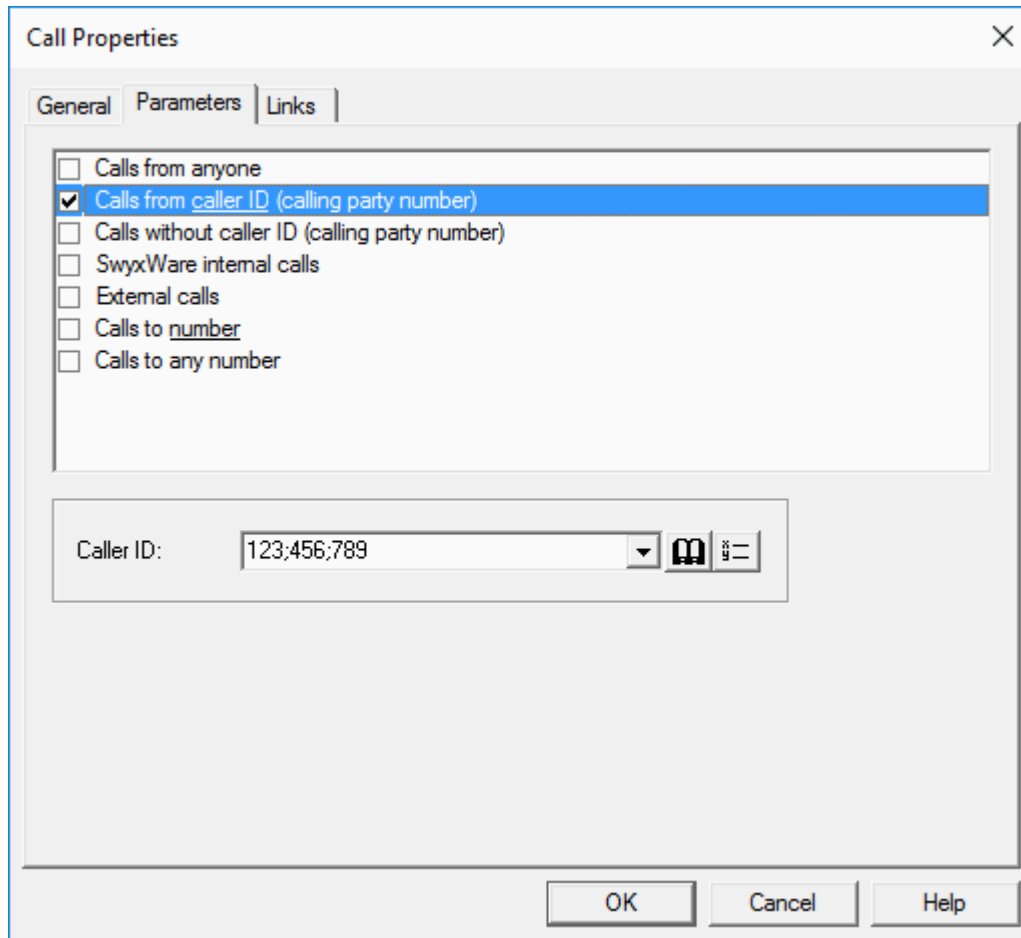
By leaving the rule through the **Rule skipped** exit, the following call routing will be executed (which should/will connect the caller).

To block a caller the **Terminate Call** block is used to disconnect the call. Within its properties you can define what the caller should get signalled. The examples here signal **Busy**, but its up to you, what you want to select here.

After disconnecting the call an **Insert Script Code** blocks is used, to prevent a **call journal** (phone call list) entry to be written for this blocked call. This is done by calling the Server Script API function PBXCall.PhoneCallListEntry.Delete. It will hide this call completely from the user. Of course it still appears in the SwyxWare **Call Detail Records**.



In both examples a **Call** block is used, to define the list of numbers. You can configure multiple numbers by separating them by a ; (semicolon).



For one or two numbers on the black or white list this block is certainly the one to use. But if you have to handle a larger list of phone numbers, which might also have to be edited by other users who do not have access to the Call Routing Manager you might want to think of another place where to keep that list.

Two possibilities would be for example to keep such a list in a **database** or **text file** and check against these lists from which in the call routing. The [function collection](#) here on Swyx Forum holds two functions showcasing how to check if a caller is known within a database or text file:

- [CheckCallerInDatabase](#)
- [CheckCallerInTextFile](#)

Instead of the **Call** block you can use an **Evaluate** block to call the chosen function, as they both return **true** or **false** if the caller is known. Please follow [this link](#) to learn on how to copy functions from the function collection into a GSE rule and how to call them afterwards in there.

As a next step you might want to make these back/white list call routing globally available for all users in your SwyxWare. In general call routing rules are always **LOCAL** to a user and can't be made **GLOBAL**. One exception of that rule is the so called **PreProcessing**. In the next blog post I will tell you all there is to know about this special type of GSE call routing rule.

**Please note:** this blog post only covers **VBScript based** call routing as of the time of writing this article the current version of **Lua based** call routing (still in beta state) does not include database access and I wanted to have this article as complete as possible. Once database access will become available for Lua based call routing I will provide the missing information to this post.

Black List (VBScript based)



**BlackList.rse**

1.54 kB · 168 downloads

White List (VBScript based)



**WhiteList.rse**

1.55 kB · 44 downloads

Enjoy!

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

< Previous entry  
#20: How to place your own VBScript/Lua code int...

Next entry >  
#22: Global call routing rules? Meet the "PreProces...

## 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