Hi

I have been using a technique (I don't know if it is really new or not, but it is not listed on this page yet) to store StatusGlobals (like AllowAutoEnter, ScriptTriggersBlocked) etc.

I got inspired by Matt Petrowsky's idea to use custom functions for more than just storing a simple calculation. At a certain point in a project I ran into the problem that in one of my scripts I turned the scripttriggers off ($$blockScriptTriggers = 1) and forgot to turn it off at the end of the script ($$blockScriptTriggers = 0).

This meant that in certain conditions the application simply failed... no more OnRecordLoad, OnLayoutLoad etc...

As it was really a pain in the ass to find the script that caused it, I came to the idea that I actually wanted to know which script lastly modified this statusGlobal. So I created for this "ScriptTriggersBlocked" status three different custom functions (SG is my "class" name for all the StatusGlobals):

- SG_ScriptTriggersLockedSet (_boolean)
- SG_ScriptTriggersLockedGet -> returns a boolean
- SG_ScriptTriggersLockedInfo -> returns more detailed info about the last modification of the status global

The custom functions uses xmlSet (_tag : _data) and xmlGet (_tag : _data) developed by Fabrice Nordmann, they can be downloaded here:


SG_ScriptTriggersLockedSet (_boolean)

Let ($$SG_ScriptTriggersLocked = 
xmlSet ("status" : GetAsBoolean (_boolean)) & 
xmlSet ("scriptIdName" : Get (ScriptName)) & 
xmlSet ("CurrentTime" : Get (CurrentTimeStamp)) & 
xmlSet ("LayoutName" : Get (LayoutName));
"
)

SG_ScriptTriggersLockedGet

Let (_status = xmlGet ($$SG_ScriptTriggersLocked : "status")
;
GetAsBoolean (_status)
)

SG_ScriptTriggersLockedInfo

$$SG_ScriptTriggersLocked

Those custom functions saved me many times. If there is now a particular strange behavior with my scripttriggers, I just put in my dataviewer SG_ScriptTriggersLockedInfo and I get the information of the latest status change.

I use the same principle for "Auto-enter block", "Turn Audit On", etc etc

For me this is really something that has an added-value for every developer, but as I said in the beginning of my post, I don't know if this is a known technique or not... Or maybe there are better techniques...