Deploying macros
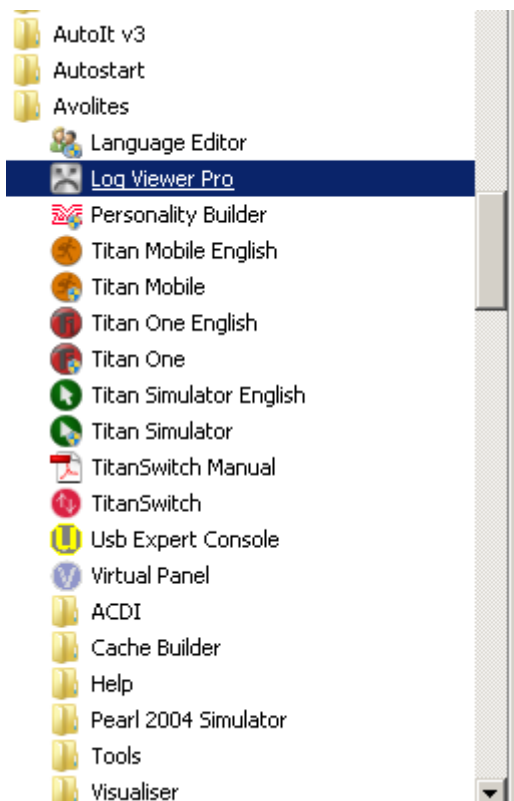
Now you have written your groundbreaking macro… but: how to make Titan aware of it and use it?
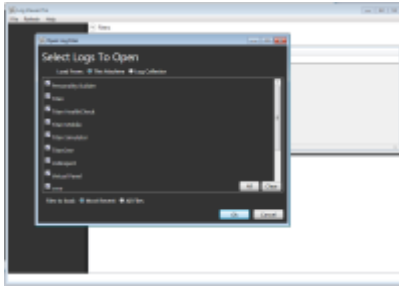
All you need to do is

- copy the macro file to your [usermacro location](#)
- restart Titan - macro files are only read upon software start
- after the software has come back on again you may open the show library (or select `<Macro>` `[View All]`) - here you'll find your new macro in the Unassigned section.
- you may also find your macro by searching it: `<Macro>` `[Play]`, and type in the first letter of your macro's name.
- if you need to call your macro more frequently, you may copy it to almost any handle you might find suitable (buttons, keys, …)
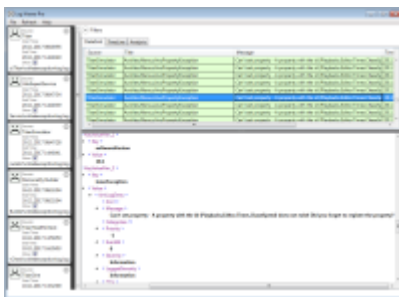
Debugging macros

Unfortunately there is not much you can do in terms of debugging a macro. However, at some point you might want to get accustomed to using the Log Viewer Pro which is installed together with the PC suite:

This opens as a mostly empty window. You need to select 'Open Folders' from the File menu.

For a quick check you can just use the default options ('This Machine', all options checked, 'Most Recent'), and hit 0k . This opens a list of the most recent events which have been logged. At least it is worth a try to check if there are entries e.g. referring to scripting errors, like this one (hint: sort by date/time):

This error says

Can't set property - A property with the id (Playbacks.Editor.Times.ChaseSpeed) does not exist! Did you forget to register the property?

- and indeed this property which was present in Titan 9.1 is not available in Titan 10.1, see Chase - Double speed

Altering macros when used in a show

tbd

I tend to think you need to delete the handle and re-copy it, or something similar, in order to take over any changes.

further readings

- Introduction to macros
- Console and simulator - how actions on the consoles are described
- Recorded vs. coded macros - both kinds: Country, AND Western
- Macro file format - what to observe when creating macro files
- Macro Folders - where exactly are the macro files stored
- Deploying macros - how to import a macro file into Titan
- XML format - a veeeery basic introduction into the format macro files are written in
- The Syntax of Functions - understanding how functions are described in general
- Control Structures - conditions and other means to control the flow
- Action and Menus - when a menu needs to be toggled in addition to the action
- Step Pause - a little delay might sometimes be helpful
- Active Binding - highlighting a macro handle as active

- [Namespaces](#) - a way to keep order of the functions, properties and other stuff
- [Datatypes](#) - numbers, words, yes & no: the various types of values
- [Properties list](#) - the affected system variables of Titan
- [Function list](#) - the functions mentioned in this wiki
- [Examples list](#) - all the contributed macros. And where is yours?

2017/10/13 15:12 · icke_siegen