

Future plans - Pavel

Contributed by Pavel
Wednesday, 21 January 2009
Last Updated Wednesday, 21 January 2009

This blog entry describes what I intend to do in the near future. It does not collide with our roadmap, there will be only one new item inside (scripting), and the implementation will be planned carefully in order to exclude any interference with other functionality.

O.K., so what's inside:

- improvement of Timers plugin:
 - relative timer already implemented (ticking in a regular interval, but unsynchronized to real time)
 - absolute timer (ticking at a given time in a given interval - for example every second sync to 0:00, or every hour sync to 00:04:30, or every Monday at 8:00, or every last day of a month at 23:59:59, etc....).

- improvement of MODBUS plugin:
 - exception handling
 - device failure information in a dedicated channel
 - channel conversion when reading from MODBUS registers for various device stored types (uint16, int16, uint32, int32, float, string)
 - serial interface Master
 - implement TCP and Serial Slave, but this not in a near future

- adding of Scripting infrastructure. I think it will radically improve flexibility and usability of FS2.
 - scripting using IronPython (now matured technology in version 2.0)
 - script editing (in future also debugging) using Scintilla (Scintilla.NET wrapper)
 - script storing in Project file
 - various types of scripts, at the time I think about three types:
 - 1) "Channel" scripts - can read any channel value, make computation and write to any channel value (preferably simulation generic channels as internal variables). Activation of these scripts via configured channels' "value changed" event (that means also possible "cyclic" behavior using timer channels, or "spontaneous" (computation when input value changes) using value source channels).
 - 2) "Action" scripts for schemas: A new "Script" Action object will be defined, which will call the script according to defined channel change event. Script also becomes the source FrameworkElement object reference as a parameter, so it can freely manipulate with graphical representation of the actioned object.
 - 3) "Designer" scripts - user scripts for design time extensions for Designer, in some form of a "plugin" infrastructure, usable for writing of import or export filters for database or schemas etc. This type must be still thought over, at the time I have no closer idea how to do it.

Some next ideas for development

- New communication plug for SNMP device monitoring
- New communication plug for system resource monitoring will be I hope covered by Lada.

That's all folks.

Pavel