

TclOO, itcl-ng and the current OO

There has been a long history on object orientation (OO) in Tcl, starting with incr Tcl/itcl.

There are some object oriented packages in Tcl, the most common ones I found are:

- incr Tcl/itcl, C-implemented
- snit: Snit Is Not Itcl, Tcl coded
- XOTcl: eXtended Objectoriented Tcl, C-implemented
- TclOO, C-implemented
- stoop: Simple Tcl Only Object Oriented Programming, Tcl coded
- ClassyTcl, C-implemented + Tcl coded

From my point of view, the still maintained and used ones are: itcl, snit, XOTcl and TclOO.

There has been a lot of discussion if OO should be integrated into Tcl core, starting with Tip#50 see: <http://tip.tcl.tk/50> , which was voted with yes in 2001, but was then put to deferred again.

A lot of time later Tip#257 (TclOO) see <http://tip.tcl.tk/257> was created as a common base for future OO extensions. Soon after that Tip#279 (TMOS) see <http://tip.tcl.tk/279> was created from the XOTcl side as a different common base for future OO.

There has been a lot of requests from the Tcl community to have „one object oriented system“ in Tcl core, as most of the other scripting languages also have one.

Concerning TclOO and TMOS there was a lot of controversial discussion, which one should be preferred. Especially when the call for vote on TclOO was announced to be soon and during the vote. Finally Tip#257 (TclOO) made it, as it has been accepted on Friday 30th Mai, to be integrated into Tcl core. There was a vote of 5 yes and 0 no from the 16 TCT members. The integration is already mostly done for Tcl 8.6a1. And TclOO will definitely be integrated into Tcl core for Tcl 8.6 to be released in September ***YES September 2008!!!***

There is a next generation implementation started for itcl (itcl-ng) see <http://wiki.tcl.tk/itcl-ng>, which is based on TclOO as a base and – as far as I know – there are also some efforts on using TclOO as a base for snit. Additionally Tip#308 TDBC Tcl database connectivity is being implemented using TclOO as a base.

My personal comments on TclOO <-> TMOS:

I had a deep look into the sources of TclOO and into the sources of a reference implementation of TMOS based on XOTcl 1.5. I have also implemented an itcl version mostly complete on both code bases. After all that efforts my feeling is: TclOO as a code base is much more readable and therefor maintainable, as the source code is very good documented inline and the coding style is very similar to the rest of the Tcl core. Looking into the reference implementation code of TMOS I had a lot of

problems understanding that code and I was not able – even with a deeply modified version – to make an itcl implementation based on that base running completely compatible to the current itcl version. With TclOO I was able to run an itcl implementation based on it completely. Also from the stacking level of function calls TclOO was using a much smaller stacking level (nesting of C-function calls).

During the implementation of itcl-ng I had a very close contact to the developer of TclOO (Donal K. Fellows), who did a lot of enhancements and additional interfaces for itcl-ng specific needs.