TPT Severe Issues

Introduction

The following document contains a list of known severe issues of TPT. By severe issues we mean issues/bugs in particular versions of TPT that:

1. might cause malfunctions in the behavior of TPT
2. are hard or even impossible to find by the TPT user herself/himself
3. cause the risk that bugs/defects in a SUT (system under test) are not detected by TPT in cases where TPT would have been able to reveal these bugs/defects in the SUT without the aforementioned malfunction in the behavior of TPT.

Usually there severe issues address the situations where the problem might appear and have well-defined workarounds.

ISSUE # 26884

TITLE:

Local search and replace in step lists always replaced all occurrences in table steps at once and so may replace entries unnoticeably the user did not intend to change.

ISSUE DETECTION:

15-Jul-19

AFFECTED VERSIONS OF TPT:

TPT 12 - TPT 13

PRECONDITIONS:

A step list including a table step with multiple occurrences of a search term and trying to replace one of these.

DETAILS:

If a search term occurs multiple times in a table step trying to replace one of them using the search dialog always replaces all occurrences in the table step. Generally, the issue affects all tables embedded in other tables but the table step is the only one where this condition applies.

EFFECT OF THE ISSUE:

Due the unexpected behavior of search and replace more changes than intended may be done to table steps leading to unintended TPT model behavior.

WORKAROUND:

Using global search and replace to replace occurrences of a search term in table steps. Changes done by local search and replace affecting table steps have to be reviewed carefully.

RESOLVED IN:

TPT12u3, TPT13u2

ISSUE # 26796
==========
TITLE:

If explicit array-of-struct values are being used in TPT models (e.g. “mychannel := [{2,3},{4,param+5}]”) these expressions can compute wrong values at runtime if the array-of-struct expression refers to non-constant channels or parameters.

ISSUE DETECTION:

01-Jul-19

AFFECTED VERSIONS OF TPT:

TPT 12 - TPT 13

PRECONDITIONS:

TPT computes wrong values at runtime under the following conditions:

- an explicit array-of-struct value is assigned to a channel or parameter (e.g. in a step list) AND
- the array-of-struct value itself depends on channels/parameters AND
- these channels/parameters are not constant (at runtime) AND

- these channels/parameters have values that differ from their default values (as specified in the declaration editor)

DETAILS:

In the special case of explicit array-of-struct expressions (e.g. “mychannel := [{2,3},{4,param+5}]”) any reference to channels/parameters inside this expression will not be computed dynamically (at runtime), but computed just once when initializing the expression (before runtime). Therefore, if the value of channels/parameters that are referred inside this expression changes at runtime, the array-of-struct expression will not update its value.

The fix now explicitly prohibits the usage of

(a) references to all channels

(b) references to all parameters that are modified dynamically in the test model

inside any explicit array-of-struct expression. Otherwise the TPT compiler will raise a compile error.

EFFECT OF THE ISSUE:

In the special case of explicit array-of-struct expressions (e.g. “mychannel := [{2,3},{4,param+5}]”) with references to non-constant channels or parameters, the value of the array-of-struct expression gets stuck with its initial value for all points in time even if the referred channels/parameters change.

After the fix related to this issue, those expressions are denied by the compiler.

WORKAROUND:

Instead of assigning the whole array-of-struct use individual assignments for non-constant elements.

RESOLVED IN:

TPT12u3, TPT13u2