

4502, 4710: PTE PAWS Test Executive Release 3.9.7 (19980714) Release Notes

Table of Contents

- 1.0 Overview
 - 1.1 Warning to CEM Users
 - 1.2 Enhancements
 - 1.2.1 RTS and CEM
 - 1.2.2 PTE
 - 1.3 Problem Reports
- 2.0 Detailed Description
 - 2.1 Enhancements
 - 2.1.1 Support for large data transfers between RTS and CEM
 - 2.1.2 Support for new DAT (Date) AIL
 - 2.2 Problem Reports

1.0 Overview

The following describes an overview of changes included within version 19980714 (3.9.7) of the PTE PAWS Test Executive. PTE includes both the Run-Time System (RTS) and the Simulator (SIM) unless otherwise indicated.

1.1 Warning to CEM Users

This version of the RTS is NOT compatible with CEM Modules built with the CEM Files distributed with previous RTS versions. After installation of this version of the RTS, you MUST recompile and relink your CEM Module(s) with the TYX-supplied CEM Files distributed with this version of the RTS.

1.2 Enhancements

1.2.1 RTS and CEM

RTS enhanced to support large data transfers between RTS and CEM.

1.2.2 PTE - DAT (Date) AIL

PTE enhanced to support anticipated change to DAT (Date) AIL to be produced by IEEE 716-89 ATLAS Compiler.

1.3 Problem Reports

Corrections for the following Problem Report were incorporated in this Release:

PR 98-035 "PTE fails to return milliseconds in response to ATLAS DATE Function."

2.0 Detailed Description

2.1 Enhancements

2.1.1 Support for large data transfers between RTS and CEM

Prior to this release, the RTS and CEM Modules had a single level of buffering for the transmission of an RTS/CEM Message. Additionally, the CEM Module called the TPS OS Memory Functions (*OSmalloc()* and *OSmfree()*) to acquire and release memory for the purpose holding the data for each ATLAS Array Element.

For UNIX Systems, the single level of buffering for an RTS/CEM Message did not cause a problem because the UNIX CEM Module is a separate Program and RTS/CEM Messages are transmitted via UNIX Pipes (which provide a second level of buffering). For MS Windows Systems, the CEM Module is a Dynamically Linked Library (DLL) with RTS/CEM Messages passed through the Message Buffer (4K Bytes for 16-Bit Windows Systems and 64K Bytes for 32-Bit Windows Systems), and large amounts of ATLAS Data were exceeding the size of the Message Buffer - a fatal error. Additionally, for 16-Bit Windows Systems, ATLAS Arrays with thousands of Elements were causing the CEM Module to exceed the maximum number (a few thousand) of memory allocations allowed by the 16-Bit Windows Operating System - also a fatal error.

With this release, the RTS and CEM Modules have a second level of buffering for RTS/CEM Messages. When the Message Buffer becomes full, it is buffered into dynamically allocated memory. When the Receiving Module detects a CR/LF sequence of Characters, the Receiving Module proceeds to un-buffer the Message into its Message Buffer and process the Message. Additionally, the RTS and CEM Modules no longer call the TPS OS Memory Allocation Functions - they call the TPS OS Huge Memory Allocation Functions *OShalloc()* and *OShfree()* which dynamically allocate big segments of memory from the Operating System, manage these segments, and return the amount of memory requested by the caller.

Note The allocation and management of segments is disabled by default - *OShalloc()* and *OShfree()* simply call *OSmalloc()* and *OSmfree()*, respectively. To enable the allocation and management of segments, the Environment Variable **TYXOSHSEGMENTSZIE** must be set. For 32-Bit Systems, it MUST be set to 65100. (At present, the maximum number of memory allocations allowed by a 32-Bit Operating System has not been exceeded on any UNIX or MS Windows NT Operating System.) For 16-Bit Systems, it SHOULD be set to 65100.

2.1.2 Support for new DAT (Date) AIL

The DAT AIL is generated by the IEEE 716-89 ATLAS Compiler as a result of an ATLAS Statement containing a call to the ATLAS Function DATE.

Prior to this release, PTE built the 15-Character Date String YYMMDDHHMMSSmmm and returned it to the ATLAS Program via the Result and Result-Qualifier Fields of the DAT AIL.

With this release, the PTE performs the following actions:

Builds a 17-Character Date String YYYYMMDDHHMMSSmmm using the enhanced capabilities of TPS Function *OStime()* to format the milliseconds (mmm).

If the Operand #1 Data Item is NOT an Integer:

Assumes the DAT AIL is the result of an ATLAS call to the ATLAS DATE Function.

Returns the 15-Character Date String YYMMDDHHMMSSmmm exactly as before (Note See Problem Report 98-035 in this document).

If the Operand #1 Data Item IS an Integer:

Assumes the DAT AIL is the result of an ATLAS call to the (proposed) ATLAS Function TIME-DATE, and this Data Item, along with the next six Data Items, are expected to be an ATLAS Record TIME-DATE-REC into which PTE will store data and is declared in ATLAS as follows:

```

DECLARE, TYPE, 'TIME-DATE-REC' IS RECORD OF
{
  'TD-YEAR'    IS INTEGER SUBRANGE 1950 THRU 2200;
  'TD-MONTH'  IS INTEGER SUBRANGE  1 THRU  12;
  'TD-DAY'    IS INTEGER SUBRANGE  1 THRU  31;
  'TD-HOUR'   IS INTEGER SUBRANGE  0 THRU  23;
  'TD-MIN'    IS INTEGER SUBRANGE  0 THRU  59;
  'TD-SEC'    IS INTEGER SUBRANGE  0 THRU  59;
  'TD-SEC-FRAC' IS DECIMAL SUBRANGE 0.0 THRU  1.0
} $
DECLARE, VARIABLE, 'CURRENT-TD' IS 'TIME-DATE-REC' $

```

Stores the appropriate binary values (acquired and converted from the 17-Character Date String) into the appropriate Data Items.

Returns the 17-Character Date String to the ATLAS Program.

Note The currently released IEEE 716-89 ATLAS Compiler does NOT support the ATLAS TIME-DATE Function. It is anticipated that this capability will be provided in the next release of this ATLAS Compiler.

2.2 Problem Reports

PR 98-035 "PTE fails to return milliseconds in response to ATLAS DATE Function."

Prior to this release, the PTE returned a 15-Character Date String YYMMDDHHMMSSmmm (where mmm is the number of milliseconds) with mmm always equal to "000".

With this release, the PTE returns the same 15-Character Date String with the number of milliseconds appropriately set.