

**Release Notes**  
**RTS Version 3.9.16**  
**28 November 1999**

1.0 Overview

This document describes changes included within:

Module Name	Module Description
SysConf	System Configuration
TPS	TYX Programming Support
PORF	PAWS Output Report Formatter
Targetter	ATLAS Program Targetter
WinUtil	TYX MS Windows Utilities
PLI	PAWS LAPS Interpreter
PTE	PAWS Test Executive

**Note** SysConf is used by TPS, PORF, Targetter, WinUtil, PLI and PTE.  
 TPS is used by PORF, Targetter, WinUtil, PLI and PTE.  
 WinUtil is used by PLI and PTE.

<u>Module Changes</u>	<u>Rebuilt</u>	<u>Current Version</u>
SysConf	None	N/A 19990610
TPS	None	No 19990803
TPS 04D	None	No 19991026
PORF	None	No 19990610
Targetter	None	No 19990610 3.8.3
WinUtil	None	No 19990610
PLI	Minor	Yes 19991128 3.9.10
PTE	Major	Yes 19991128 3.9.16

PTE means both the Run-Time System (RTS) and the Simulator (SIM).

PTE, RTS, CEM and SIM mean all Platforms.

UPTE, URTS, UCEM and USIM mean UNIX Platforms only.

WPTE, WRTS, WCEM and WSIM mean MS-Windows Platforms only.

XPTE, XRTS and XSIM mean X-Window Platforms only.

**NOTICE TO CEM USERS**

This version of the RTS is compatible with CEM Modules built with CEM Files distributed with RTS Version 19980714 3.9.7 and later. You may install this version of the RTS without rebuilding your CEM Module. However, in order to make use of the new CEM capabilities, you **MUST** install this version of the RTS, and then you **MUST** recompile and relink your CEM Module(s) with the TYX-supplied CEM Files distributed with this version of the RTS.

As time goes on, RTS/CEM testing to provide error-free backwards-compatibility becomes increasingly difficult. Therefore, TYX Corporation very strongly recommends that CEM Users recompile and relink their CEM Modules after installing a new version

of the RTS.

1.1 Enhancements

1.1.1 System Configuration – None.

1.1.2 TYX Programming Support - None.

1.1.3 PAWS Output Report Formatter - None.

1.1.4 ATLAS Program Targetter - None.

1.1.5 TYX Windows Utilities - None.

1.1.6 PAWS LAPS Interpreter - Minor.

Support for ASCII Plus Sign in Lexical Tokens.

1.1.7 PAWS Test Executive - Major.

CEM Module.

Fix for CEM Kernel Fault.

Change to CEM Kernel Data Reset.

Elimination of **V\_INT** Multiple Definition.

Changes to User Stub Functions:

New Values for Environment Variable **TYXCEMSTUBLEVEL**.

New Printing Algorithm for User Stub Functions.

Change to Action Lines for Non-ATLAS Actions.

Change to Modifier Data Lines for FETCH and LOAD ATLAS Actions.

New User Stub Debug Functions *userStubPrintOnly()* and *userStubSetLevel()*.

1.2 Problem Reports - None.

## 2.0 Detailed Description

### 2.1 Enhancements

2.1.1 System Configuration - None.

2.1.2 TYX Programming Support - None.

2.1.3 PAWS Output Report Formatter - None.

2.1.4 ATLAS Program Targetter - None.

2.1.5 TYX Windows Utilities - None.

2.1.6 PAWS LAPS Interpreter - Minor.

2.1.6.1 Support for ASCII Plus Sign in Lexical Tokens.

Prior to this release, PLI did not support ASCII Plus Signs (+) in Lexical Tokens.

With this release, PLI supports ASCII Plus Signs (+) in Lexical Tokens.

## 2.1.7 PAWS Test Executive - Major.

### 2.1.7.1 CEM Module.

#### 2.1.7.1.1 Fix for CEM Kernel Fault.

Prior to this release, the CEM Kernel would sometimes fault (or cause other unpleasant things to occur later) because of an error during the initial processing of Modifiers with a LEVEL-LOGIC Prefix.

With this release, this error has been corrected.

#### 2.1.7.1.2 Change to CEM Kernel Data Reset.

Prior to this release, the CEM Kernel would reset all Datums for the Current Device / Channel before calling the User's RESET Function during the IFC/RESETs/DCL Loop. This was inconsistent with the processing that occurred for an ATLAS Action RESET, where the CEM Kernel resets all Datums for the Current Device / Channel after calling the User's RESET Function.

With this release, the CEM Kernel resets all Datums for the Current Device / Channel after calling the User's RESET Function during the IFC/RESETs/DCL Loop.

#### 2.1.7.1.3 Elimination of **V\_INT** Multiple Definition.

Prior to this release, compilation of CEM User Source Files using certain versions of the Microsoft C++ Compiler would print a Warning Message indicating that Label **V\_INT** was multiply defined.

With this release, the CEM Header File **cem.h** contains Compiler Directives to undefine **V\_INT** if it is defined, thereby eliminating the Warning Message.

2.1.7.1.4 Changes to User Stub Functions.

2.1.7.1.4.1 New Values for Environment Variable **TYXCEMSTUBLEVEL**

Prior to this release, the numerical value assigned to **TYXCEMSTUBLEVEL** caused User Stub Functions to print Action Lines and Modifier Data Lines as follows:

<u>Value</u>	<u>Meaning</u>
0	No printing.
Not 0	Print Action Lines and Modifier Data Lines.

With this release, the numerical value assigned to **TYXCEMSTUBLEVEL** causes User Stub Functions to print Action Lines and Modifier Data Lines as follows:

<u>Value</u>	<u>Meaning</u>
0	No printing.
1	Print Action Lines.
2	Same as 1 above, plus Modifier Data Lines for all Datums that have not yet been printed.
3	Same as 2 above, plus Modifier Data Lines for all "stale" Datums detected by the User Stub SETUP Function <i>userStubSETUP()</i> .
4	Same as 3 above, plus Modifier Data Lines for all "stale" Datums detected by the User Stub CLOSE Function <i>userStubCLOSE()</i> .
5	Same as 4 above, plus Modifier Data Lines for all "stale" Datums detected by the User Stub INITIATE Function <i>userStubINIT()</i> .
6	Same as 5 above, plus Modifier Data Lines for all "stale" Datums detected by the User Stub FETCH Function <i>userStubFETCH()</i> .
7	Same as 6 above, plus Modifier Data Lines for all "stale" Datums detected by the User Stub LOAD Function <i>userStubLOAD()</i> .
8	Same as 2 above, plus Modifier Data Lines for all "stale" Datums.
9	Modifier Data Lines for all Datums.

Refer to the next two Sections for more information pertaining to the usage of these values.

2.1.7.1.4.2 New Printing Algorithm for User Stub Functions.

Prior to this release, certain User Stub Functions printed Modifier Data Lines for certain Datums, depending upon the Datum Qualifier. For example, *userStubSETUP()* printed Modifier Data Lines only for Datums with Setup Qualifiers, *userStubCONNECT()* printed Modifier Data Lines only for Datums with Connect Qualifiers, *userStubFETCH()* printed a Modifier Data Line for the Fetch Datum, etc. Subsequent testing with various ATLAS Compilers has shown this to be insufficient.

With this release, User Stub Functions print Modifier Data Lines as defined by the

numerical value assigned to the Environment Variable **TYXCEMSTUBLEVEL** as defined in the previous Section. Additionally, some User Stub Functions (e.g., *userStubINIT()* and *userStubFETCH()*) that did not print Modifier Data Lines for Datum with Setup Qualifiers now do so.

#### 2.1.7.1.4.3 New User Stub Debug Functions *userStubPrintOnly()* and *userStubSetLevel()*.

Prior to this release, a User Function within an existing CEM Device Driver that included a call to the appropriate User Stub Function in order to get Modifier Data Lines printed ran into problems. Due to the old printing algorithm (see previous Section), not all Datums could be seen. Additionally, if *userStubFETCH()* was called, the Fetch Datum was loaded with random Datum Values.

With this release, two new User Stub Debug Functions have been added to support CEM Device Driver debugging:

<u>Function Prototype</u>	<u>Description</u>
<code>void userStubPrintOnly( void )</code>	Requests that the next User Stub Function called only perform printing processing. As currently implemented, the only User Stub Function affected is <i>userStubFETCH()</i> .
<code>void userStubSetLevel( int nSL )</code>	Requests that the next User Stub Function called perform printing at the specified Stub Level <b>nSL</b> , thereby temporarily overriding the setting of <b>TYXCEMSTUBLEVEL</b> .

Both of these Functions are "one-shot" Functions in that the request applies to the next User Stub Function called. The following are examples of usage:

```
SomeDeviceSlashChannelINIT()
{
    userStubSetLevel( 9 );    /* Request all Data
    */
    userStubINIT();         /* Print Action and Data
    */
    (User's Processing)
}

SomeDeviceSlashChannelFETCH()
{
    (User's Processing)
    userStubPrintOnly();    /* Disable Random Data
    */
    userStubFETCH();       /* Print Action and Data
    */
}
```

#### 2.1.7.1.4.4 Change to Action Lines for Non-ATLAS Actions.

Prior to this release, the Action Name was placed on the left-hand side of an Action Line (thereby indicating a Non-ATLAS Action) whenever the Current Verb and Current Noun were zero. This produced incorrect information when the RTS executed an ATLAS Terminate Statement.

With this release, Action Lines for Non-ATLAS Actions are correctly formatted regardless of the Current Verb and Noun. Additionally, the Action Line printed by *userStubRESET()* during the IFC/RESETs/DCL Sequence shows the appropriate Controller and Device Bus Addresses.

#### 2.1.7.1.4.5 Change to Modifier Data Lines for FETCH and LOAD ATLAS Actions.

Prior to this release, the User Stub Function *userStubFETCH()* and *userStubLOAD()* printed Modifier Data Lines only for the Fetch and Load Datums, respectively. These Lines always started with "M1". This became confusing when these Functions were changed to also print Modifier Data Lines for Datums with Setup Qualifiers.

With this release, the Modifier Data Lines for the Fetch and Load Datums begin with "MF" and "ML", respectively.