

**PAWS**  
**Release Notes - Miscellaneous**  
**7/23/98**

**Table of Contents**

- 1.0 Overview
  - 1.1 Enhancements
    - 1.1.1 TYX Programming Support
      - 1.1.1.1 Function *OStime()*
- 2.0 Detailed Description
  - 2.1 Enhancements
    - 2.1.1 TYX Programming Support
      - 2.1.1.1 Time-of-Day down to number of milliseconds

**1.0 Overview**

The following describes an overview of miscellaneous changes included with this release of software. The changes included in these Release Notes were made at the same time as the changes covered in PAWS Test Executive, Release 3.9.7 . PTE includes both the Run-Time System (RTS) and the Simulator (SIM).

This document describes changes included within Module:

<u>Module Name</u>	<u>Version</u>
TPS: TYX Programming Support	19980714

Note: TPS is used by PTE.

**1.1 Enhancements**

**1.1.1 TYX Programming Support**

**1.1.1.1 Function *OStime()***

Enhanced to provide Time-of-Day down to the number of milliseconds.

**2.0 Detailed Description**

**2.1 Enhancements**

**2.1.1 TYX Programming Support**

**2.1.1.1 Time-of-Day down to number of milliseconds.**

Function *OStime()* is the TYX interface function to the Operating System for the purpose of acquiring the current Time-of-Day in Seconds-Since-The-Epoch.

Prior to this release, *OStime()* provided the caller with the Time-of-Day down to the number of seconds acquired from the C Library Function *time()*.

With this release, *OStime()* provides the caller with the Time-of-Day down to the number of milliseconds.

The caller's interface to *OStime()* has not changed - the caller still specifies a Pointer to a Long Integer into which *OStime()* stores the Time-of-Day in seconds. *OStime()* also acquires the number of milliseconds

from the Operating System and stores this value (range:  $0 \leq n \leq 999$ ) in a new Global Integer **OSntimms**. The caller may use **OSntimms** or not, as desired.

The C Library Function *gettimeofday()* is the preferred function to be called to acquire the Time-of-Day. However, some C Libraries do not (or do not yet) support this function. For those Systems, the C Library Function *ftime()* is called.

### **CAUTION TO C PROGRAMMERS**

The current standard definition of the Epoch is January 1, 1970 at 00:00:00. When *OStime()* was defined and implemented some 10+ years ago, some C Libraries either did not support *time()* or had a different definition of the Epoch. Therefore, *OStime()* defined the Epoch as January 1, 1985 at 00:00:00, thereby defining a constant Epoch for all Platforms. When the port to the MS-DOS Platform was done a few years ago, this definition of the Epoch was NOT used - the MS-DOS *OStime()* simply returned whatever *time()* provided. C Programmers are cautioned to NOT use the absolute value provided by either *OStime()* or *time()* as a means of identifying Files because the Epoch may not be the same on different Platforms. Instead, use the TPS System Library Functions *SLatime()* and/or *SLctime()* to convert the *OStime()* value to a String of the form YYYYMMDD, and then either use that String or convert it to a binary value. It is suspected that there are some multi-platform TYX Files which contain the *OStime()* absolute value, but no investigation has been done at this time.