

**CEM User
Release Notes
Release 20030114
Version 3.9.26**

Purpose

The purpose of this document is to provide information pertaining to the release of the CEM Kernel and user interface. This document is divided into three sections:

<u>Enhancements</u>	Describes changes that enhance the capability of the CEM Kernel.
<u>Problem Reports</u>	Describes changes that correct errors reported in problem reports.
<u>CEM Help</u>	Describes changes made to the user interface that will eventually be added to the CEM On-Line Help.

Enhancements

Change from LINUX-Based YACC to Windows-Based YACC (Windows Systems only)

The RTS sends Text Messages to the CEM Kernel requesting the CEM Kernel to perform various actions. These Text Messages are parsed by the Message Syntax Processor (MSP) portion of the CEM Kernel. The MSP Source Code is written in the YACC (Yet Another Compiler Compiler) Language, which is read by a YACC Application, which in turn produces C Language Source Code.

Prior to this release, TYX Corporation did not have a YACC Application that ran on a Windows or DOS System. The YACC Application used to generate the MSP C Language Source Code was the one provided with one of TYX Corporation's LINUX Systems.

TYX Corporation has acquired a YACC Application that runs on a Windows/DOS System.

With this release, MSP C Language Source Code is generated by a YACC Application that runs on a Windows System.

Problem Reports

None

Problems Detected In-House

Macro *GetCurCtlMLA()* Sometimes Returns Invalid Address During ATLAS Action Processing

There are three protocols (all of which are currently supported by the CEM Kernel) defining how an RTS can pass Controller Addresses to the CEM Kernel:

Protocol 1 This protocol is the original one whereby the CEM Kernel extracts the Controller MTA or MLA from the RTS-to-CEM Command Message pertaining to the Current Device and saves it for use by the CEM Driver handling the Current Device. When the CEM Kernel receives a Command Message specifying a different Device, the Controller Addresses are reset.

Limitation No Controller Addresses are available until the ATLAS Program has started execution.

Protocol 2 This protocol was designed to correct the limitation of Protocol 1 by defining a “1st CEM Bus” RTS-to-CEM Information Messages that allows an RTS to send the MTA and MLA of the “First Controller” to the CEM Kernel. These Messages are sent after the Non-ATLAS CONNECT Action and before the Non-ATLAS OPEN Action, thereby making these Addresses available to a CEM Driver prior to starting execution of the ATLAS Program. The CEM Kernel saves these Addresses in the same location in which it saves the Addresses received as per Protocol 1.

Limitation The CEM Kernel correctly supports only one Controller.

Protocol 3 This protocol was to designed to correct all known Controller Address limitations by defining RTS-to-CEM Information Messages that allow an RTS to send MTA’s and MLA’s for all Controllers to the CEM Kernel.

The UNIX RTS and WMENU RTS use Protocol 3. The Studio RTS (which requires that different Controllers must be handled by different CEM Modules) uses Protocol 2.

Prior to this release, the CEM Kernel saved Controller Addresses received as per Protocol 2 only in the same location in which it saved those received as per Protocol 1. During its first ATLAS Action processing (normally SETUP) for a particular Device, a CEM Driver would see a valid Controller MTA but an invalid Controller MLA (as a result of the “different Device reset”).

With this release, the CEM Kernel saves Controller Addresses received as per Protocol 2 in unique locations that do not get reset (in addition to where it was

previously saving them).

CEM Help

None