



# Introduction

The main function of the Circuit Emulation Service Module (CESM) is to provide a constant bit rate (CBR) circuit emulation service by converting data streams into CBR AAL1 cells for transport across an ATM network. The CESM supports the CES-IS specifications of the ATM Forum.

The 8-port CESM lets you configure individual physical ports for structured or unstructured data transfer.

The most common application is legacy support for digitized voice from a PBX or video from a codec. Using circuit emulation, a company can expand its data communication network without specific voice or video cards to meet its voice or teleconferencing requirements.

This chapter introduces the CESM cards that are supported by the PXM1E card in MGX 8850 and MGX 8830 switches. This chapter also describes the features of these cards.

## CESM Card Types

When operating under control of the PXM1E card, the MGX 8850 and MGX 8830 switches support the CESM cards listed in [Table 1-1](#).

**Table 1-1 CESM Card Types Supported by PXM1E Controllers**

Front Card	Back Card	Maximum Connections
CESM-8T1/B	RJ48-8T1, RJ48-8T1-R	192
CESM-8E1	RJ48-8E1, SMB-8E1, RJ48-8E1-R, SMB-8E1-R	248

The RJ48-8T1-R and RJ48-8E1-R back cards support 1:N redundancy through the optional MGX-SRM-3T3/C card.

The preferred tools for configuring, monitoring, and controlling service modules are the CiscoView and Cisco WAN Manager applications for equipment management and connection management, respectively. However, the command-line interface (CLI) also provides access to the service modules and is highly applicable during initial installation, troubleshooting, and any situation in which low-level control is useful.

# CESM-8T1/E1 Features

Table 1-2 summarizes the key features of the CESM:

**Table 1-2 CESM card features**

Feature	CESM-8T1/E1
Service Type	Structured/Unstructured
Clocking	Sync/Async (SRTS/Adap)
Idle Supp	Yes
Partial fill	Yes
Onboard BERT	No
Redundancy	1:N

The CESM-8T1/E1

- Provides up to 8 T1/E1 interfaces.
- Allows structured (SDT) or unstructured (UDT) data transfer per physical interface.
- Provides N x 64 Kbps fractional DS1/E1 service (SDT only). In Structured Data Transfer Mode, the CESM supports fractional DS1/E1 service (time slots must be contiguous). Any N x 64 channels can be mapped to any VC. Therefore, multiple ports can be defined that are composed of unique contiguous time slots and a connection is used to emulate the data for that logical port.
- Supports synchronous timing in both Structured Data Transfer (SDT) mode and Unstructured Data Transfer (UDT) mode. Synchronous timing is derived from the switch.
- Supports asynchronous timing mode with Synchronous Residual Time Stamp (SRTS) and adaptive clock recovery methods (UDT only) can be employed for asynchronous timing.
- Provides ON/OFF hook detection and idle suppression using channel-associated signaling (CAS). Only available in SDT mode.
- Allows a choice of partially filled AAL1 cell payload to improve cell delay.
  - Type Fill Range (bytes)
  - T1 Structured: 25–47
  - T1 Unstructured: 33–47
  - E1 Structured: 20–47
  - E1 Unstructured: 33–47
- Provides a maximum number of connections: 192 T1 and 248 E1.

The following sections provide additional information on CESM features.

## CESM-8T1/E1 Peak Cell Rate Calculation

The following features are shared by all CE service modules:

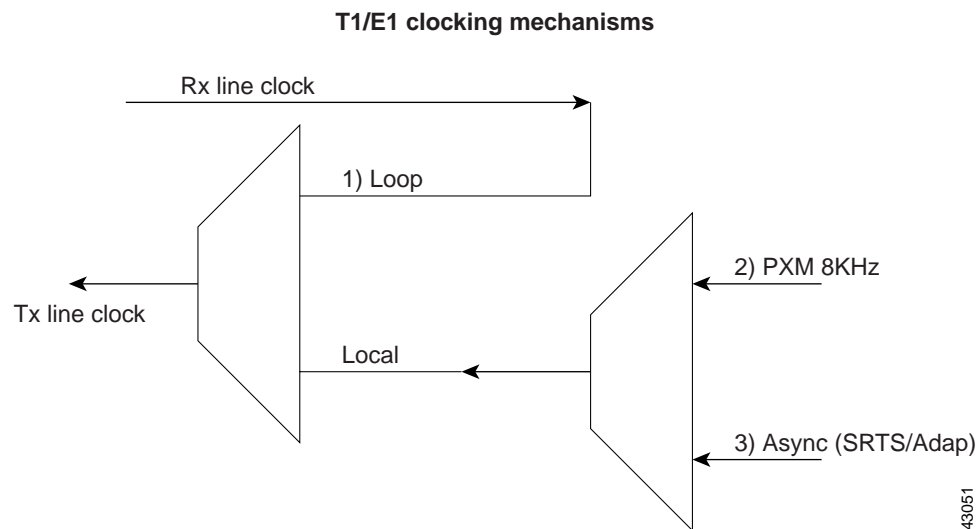
- Partial Fill Value (K bytes)
  - SDT Time slots (N channels)
  - Unstructured T1 cell rate
    - $(1.544 \times 10^6 \text{ bps}) / (\text{K octets/cell} \times 8 \text{ bits/octet})$
    - 4107 cps (for K = 47 bytes)
  - Unstructured E1 cell rate
    - $(2.048 \times 10^6 \text{ bps}) / (\text{K octets/cell} \times 8 \text{ bits/octet})$
    - 5447 cps (for K = 47 bytes)
  - Structured N x 64 cell rate, basic mode
    - $(8000 \times N) / (\text{K octets/cell})$
    - 170 cps (for K = 47, and N = 1)

## T1/E1 Clocking Mechanism

The CESM card provides the choice of a physical interface Tx clock from one of the following sources, as illustrated in [Figure 1-1](#):

1. Loop clocking derived from Rx Line Clock.
2. MGX local switch clock derived on the PXM (Synchronous).
3. SRTS and Adaptive based clock (for T1/E1 unstructured asynchronous mode only).

**Figure 1-1 T1/E1 Clocking Mechanisms**

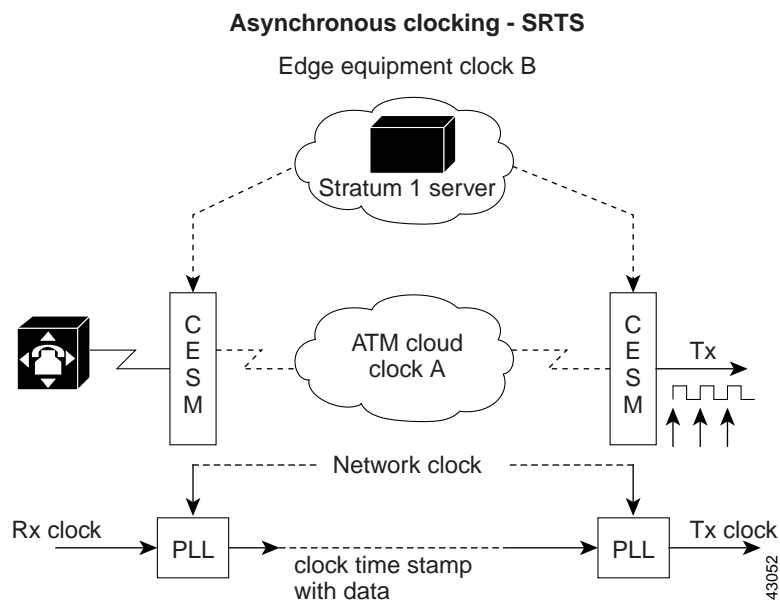


## Asynchronous Clocking (SRTS)

Synchronous Residual Time Stamp (SRTS) clocking requires a Primary Reference Source (PRS) and network clock synchronization services. This mode allows user equipment at the edges of an ATM network to use a clocking signal that is different (and completely independent) from the clocking signal being used in the ATM network. However, SRTS clocking can only be used for unstructured (clear channel) CES services.

For example, as illustrated in [Figure 1-2](#), user equipment at the edges of the network can be driven by clock B, while the devices within the ATM network are being driven by clock A. The user-end device introduces traffic into the ATM network according to clock B. The CESM segments the CBR bit stream into ATM cells; it measures the difference between user clock B, which drives it, and network clock A. This delta value is incorporated into every eighth cell. As the destination CESM receives the cells, the card not only reassembles the ATM cells into the original CBR bit stream, but also reconciles the user clock B timing signal from the delta value. Thus, during SRTS clocking, CBR traffic is synchronized between the ingress side of the CES circuit and the egress side of the circuit according to user clock B, while the ATM network continues to function according to clock A.

**Figure 1-2 Asynchronous Clocking**



## Asynchronous Clocking (Adaptive)

Adaptive clocking requires neither the network clock synchronization services nor a global PRS for effective handling of CBR traffic. Rather than using a clocking signal to convey CBR traffic through an ATM network, adaptive clocking infers appropriate timing for data transport by calculating an average data rate for the CBR traffic. However, as in the case with SRTS clocking, adaptive clocking can be used only for unstructured (clear channel) CES services. See [Figure 1-3](#).

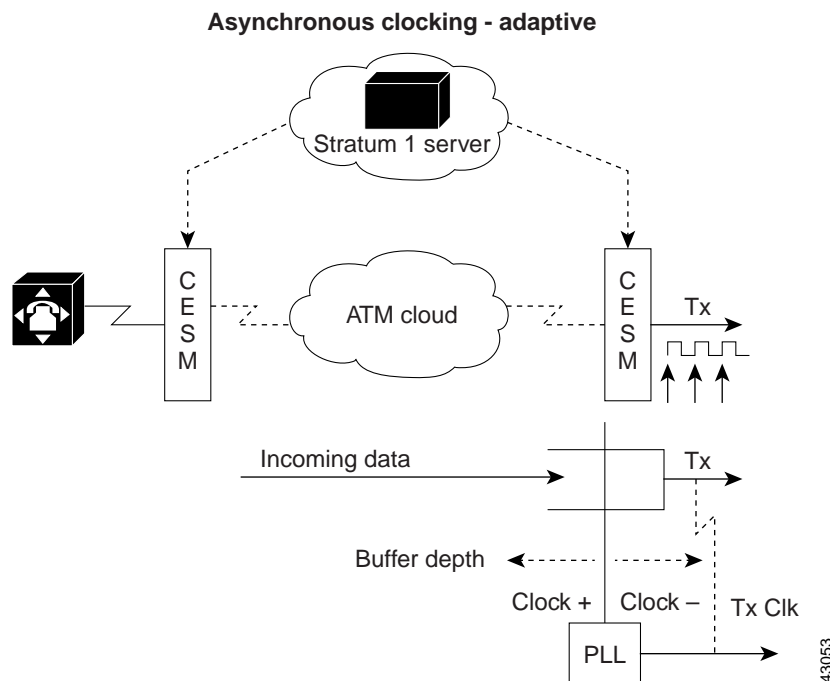
For example, if CBR data is arriving at a CES module at a rate of X bits per second, then that rate is used, in effect, to govern the flow of the CBR data through the network. What happens behind the scenes, however, is that the CES module automatically calculates the average data rate. This calculation occurs dynamically as user data traverses the network.

When the CES module senses that its segmentation and reassembly (SAR) buffer is filling up, it increases the rate of the (TX) clock for its output port, thereby draining the buffer at a rate that is consistent with the rate of data arrival.

Similarly, the CES module slows down the transmit clock of its output port if it senses that the buffer is being drained faster than the CBR data is being received. Adaptive clocking attempts to minimize wide excursions in SAR buffer loading, while at the same time providing an effective means of propagating CBR traffic through the network.

Relative to other clocking modes, implementing adaptive clocking is simple and straightforward. It does not require network clock synchronization services, a PRS, or the advance planning typically associated with developing a logical network timing map. However, adaptive clocking does not support structured CES services, and it exhibits relatively high wander characteristics.

Figure 1-3 Asynchronous Clocking (Adaptive)



## CESM Idle Suppression

The CESM T1/E1 card in structured mode can interpret CAS robbed bit signaling for T1 (ABCD for ESF and AB for SF frames) and CAS for E1 (time slot 16). The ABCD code is user configurable per VC (`xnfcchan onhkcd = 0-15`; ABCD = 0000 = 0 ... ABCD = 1111 = 15). By detecting on-hook/off-hook states, AAL1 cell transmission is suppressed for the idle channel; thereby reducing backbone bandwidth consumed. ON/OFF hook detection/suppression can be enabled/disabled per VC and ON/OFF hook states can be forced via SNMP through the NMS.

On the ingress end, the CESM card monitors the signaling bits of the AAL1 cell. Whenever a particular connection goes on-hook and off-hook, the CESM card senses this condition by comparing ABCD bits in the cell with pre programmed idle ABCD code for that channel.

When an on-hook state is detected, keep-alive cells are sent once every second to the far-end CESM. This prevents the far end from reporting an under-run trap during idle suppression, since no cells are transmitted. When the time slot switches to off-hook mode, the CESM stops sending the keep-alive cells.

## Structured Data Transfer

If you configure an individual port for structured data transfer, the 8-port CESM supports:

- Synchronous timing.
- Superframe or Extended Superframe for T1.
- Fractional ( $N \times 64$  Kbps) DS1/E1 service (contiguous time slots only). You can map an  $N \times 64$  Kbps channel to any VC.
- CAS robbed bit for T1 (ABCD for ESF and SF frames) and CAS for E1 (channel 16).
- CCS channel as a transparent data channel.
- A choice of partial-fill payload sizes.
- Idle detection and suppression for 64Kbps CAS connections.
- Loopback diagnostics on a line or a connection (**addlnloop**, **tstcon**, and **tstdelay** commands).
- Bit error rate test (BERT) functionality with loopback pattern generation and verification on individual lines or logical port.

## Unstructured Data Transfer

If you configure an individual port for unstructured data transfer, the 8-port CESM supports:

- Synchronous or asynchronous timing at T1 (1.544 Mbps) or E1 (2.048 Mbps) rates. For asynchronous timing, you can select its basis as either SRTS and adaptive clock recovery.
- The special port type *framingOnVcDisconnect*. This port type prevents a remote-end CPE from going to LOF by placing a line in remote loopback mode when the CESM determines that a connection deletion or suspension occurred at the network-side ATM interface.
- Ability to detect and display a yellow alarm for the ESF framing on a T1 line.
- Loopback diagnostics on a line or a connection (**addlnloop**, **tstcon**, and **tstdelay** commands).
- Bit error rate test (BERT) functionality with loopback pattern generation and verification on individual lines.

## Cell Delay Treatment

For each connection, you can configure a tolerable variation in the cell arrival time (CDVT) according to the expected reliability of the route. The CDVT applies to the receive buffer. After an underrun, the receiver places the contents of the first cell to arrive in a receive buffer then plays it out at least one CDVT value later. For each VC, the maximum cell delay and CDVT (or jitter) are

- For T1
  - Cell delay of 48 msec
  - CDVT of 24 msec in increments of 125 microseconds
- For E1
  - Cell delay of 64 msec
  - CDVT of 32 msec in increments of 125 microseconds

## Redundancy Support for the Eight-Port CESM

The AX-CESM-8T1 and AX-CESM-8E1 can have 1:N redundancy support but with some variations between the T1 and E1 modes of operation. The type of redundancy and the type of back card are interdependent. In general:

- With an RJ48-8T1, an MGX-SRM-3T3 can provide 1:N redundancy through the distribution bus or the redundancy bus.
- With an RJ48-8E1, an MGX-SRM-3T3 can provide 1:N redundancy through the redundancy bus.

Back card requirements for the MGX-SRM-3T3 and service modules vary, as follows:

- If you are using the MGX-SRM-3T3 for *bulk distribution* of T1 channels, the CESMs do not use back cards, but each MGX-SRM-3T3/C must have an MGX-BNC-3T3-M back card.
- Bulk distribution is available for E1 operation, but only with the MGX-SRM-3E3/C. Each MGX-SRM-3E3/C must have an MGX-BNC-3E3-M back card
- If the MGX-SRM-3T3/C supports T1 or E1 1:N redundancy through the *redundancy bus* (no bulk distribution), the MGX-SRM-3T3/C does not require a back card, but the *N* CESM primary cards must have one redundant version of the back card.

## Error and Alarm Management

The CESM card provides error and alarm management at the card, line, and connection levels. For more information, see [Chapter 3, “CESM Card Management.”](#)

