

ZebOS®  
Network Platform

---

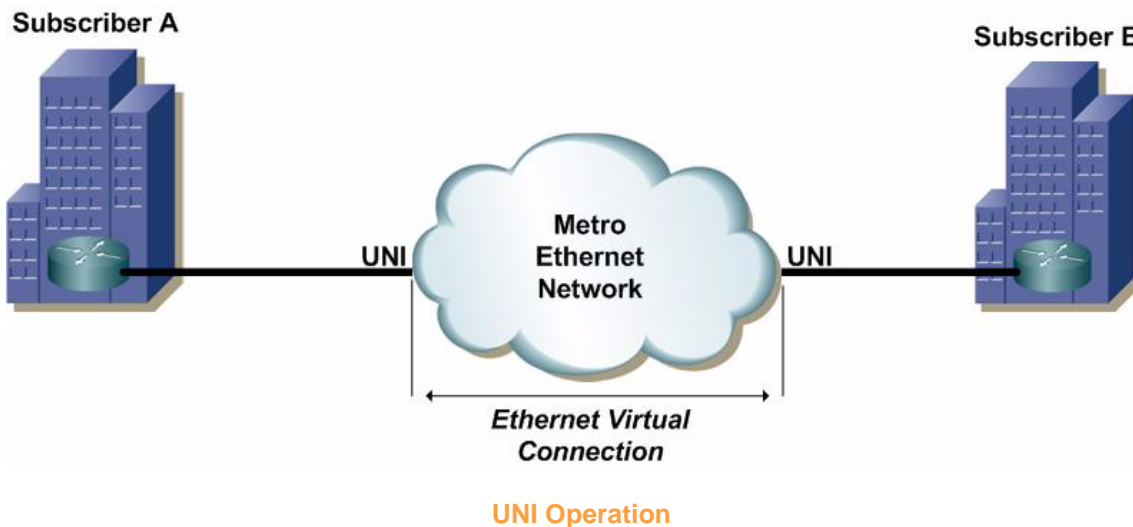
## User Network Interface Module

### Overview

The ZebOS Network Platform's User Network Interface (UNI) provides a demarcation point between the responsibility of the service provider and the responsibility of subscribers. It offers a reference point for Ethernet service delivery by establishing a standards-based connection through a Metro Ethernet Network (MEN).

Physically implemented over a bi-directional Ethernet link, UNI provides a variety of data, control and management plane capabilities required by MEN service providers. The functionality of the UNI is to separate clearly the subscribers of two network domains, which are involved in the operational, administrative, maintenance and provisioning aspects of their own network services.

The following figure displays a simple diagram of an UNI operation.



### UNI Reference Point

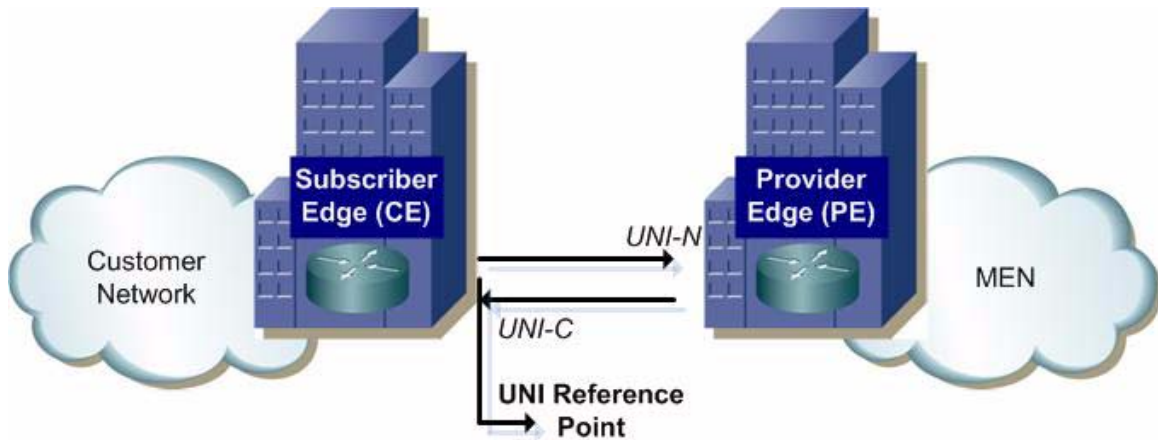
UNI is an abstract concept, with the service provider responsible for one side of the configuration and the customer responsible for the other side. The demarcation of these two points is the reference point for UNI. Reference points help service the Ethernet Virtual Circuits (EVCs) that connect the UNIs.

Typically, the reference point for UNI is located at the premises of the subscriber and acts as the demarcation point between the subscriber and the MEN boundaries. It utilizes the connector that the subscriber network uses to connect to the service provider network.

Functionally, the reference point for UNI divides the role of the subscriber (known as UNI-C) and the role of the MEN (known as UNI-N).

- UNI-N is responsible for all aspects of UNI from the Provider Edge (PE) to the reference point for UNI. Moreover, UNI-N refers to a set of functional elements that supports the technical capabilities and compliance to the specification for UNI from the MEN service provider.
- UNI-C is responsible for all aspects of UNI from the Customer Edge (CE) to the reference point of UNI. Additionally, UNI-C refers to the functional elements within the Ethernet Local Management Interface (E-LMI) that supports the technical capabilities and compliance to the specification for UNI from the MEN subscriber.

Because of the distributed nature of UNI, maintenance of both UNI-C and UNI-N is performed by different autonomies.



### UNI Detailed Operation

## UNI Functional Elements

UNI includes two functional elements that are located at the connected devices at either side of the reference point of UNI (demarcation): UNI-C and UNI-N.

- UNI-C: Responsible to execute all of the processes from the customer side
- UNI-N: Responsible to execute all of the processes from the network side

The functional interfaces of UNI-N and UNI-C do not need to be on the same physical device.

## Features

- Full implementation and support for the following:
  - Section 10 of MEF 11
  - Sections 8, 9, 10 of MEF 20
  - Sections 6, 7 and 8.4 of MEF-10.1h MEF 13 (UNI-Type1) and MEF20 (UNI-Type 2) UNI specs
- Provides a single-point of contact that is the demarcation between customers and service providers
- Located on a port of an active device that is operated by the service provider
- Easily integrated with other MEF defined functions
- Automatically configures and manages Ethernet UNI functions and services
- Complete OAM functions, including support for Connectivity Fault Management (CFM)
- Increased bandwidth and load sharing with linear increments of units of links

## Requirements

- ZebOS Network Services Manager

## Standards Supported

- MEF 2 - Requirement and Framework for Ethernet Service Protection
- MEF 11 - User Network Interface (UNI) Requirements and Framework (UNI Type 1 and 2 only). Includes:
  - MEF 20 Section 8 - UNI Type 2 Discovery and Configuration
  - MEF 20 Section 9 - Supporting E-LMI for UNI Type 2
  - MEF 20 Section 10 - Supporting Ethernet OAM (Link and Service level OAM)
  - MEF 10.1 - UNI and EVC per UNI Service Attributes
  - MEF 10.1 - EVC Service attributes
- MEF 16 - Ethernet Local Management Interface (E-LMI)
- MEF 17 - Service OAM Framework and Requirements

## Standard Deliverables

- Source Code (written in ANSI-compliant C)
- Installation Guide
- Configuration Guide
- Command Reference Guide
- Developer Guide