# TECHNOLOGY DEVELOPMENT AND SYSTEM INTEGRATION FOR A VEHICLE MILES TRAVELED BASED REVENUE COLLECTION SYSTEM PROTOTYPE

## OSU Software Documentation
## Version 1.4

## Subtask 4.2

**Prepared for**
**Oregon Department of Transportation**

Revision Date: 5/28/04

Prepared by
Department of Industrial and Manufacturing Engineering
Oregon State University

## Introduction

The OSU Software developed by the OSU research team is a critical component of both prototype technology configurations. The purpose of this document is to explain the role of the OSU Software within a technology configuration, the functions (and their timing) performed by the OSU Software, and other details specific to the current implementation. This document, along with the documented source code, should provide the information required for software developers to extend and modify the OSU Software.

## Relationship to Other System Components

The purpose of the OSU Software is to act as middleware that facilitates the communication between the DCSS POS[1] system Software, the AFX[2] Software, and the Local DB (for the odometer technology configuration) or the Central DB (for the GPS technology configuration).

The OSU Software has the following characteristics (see Figure 1):

- *Communicates with the DCSS POS system Software and the AFX Software via XML files/messages*
- *In the odometer technology configuration, it communicates with the local database management system (DBMS)*
- *In the GPS technology configuration, it communicates with the central DBMS*
- *Tracks completed transactions and processes*
- *Calculates VMT fees*

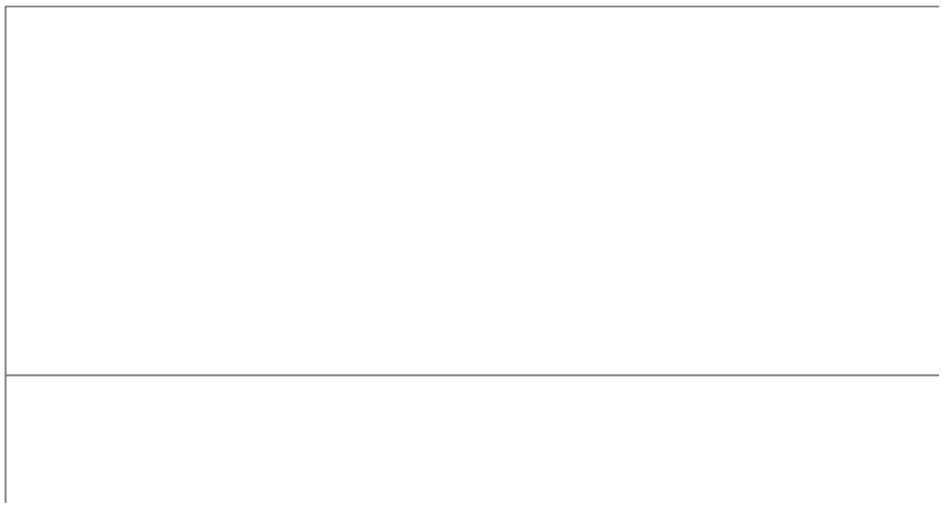

**Figure 1. Five-Tier Model of Application for the Technology Configurations.**

[1] DCSS POS = Digital Consulting and Software Services Point-of-Sale. DCSS is the company that developed a custom POS system software for this implementation.
[2] AFX = AFX Technology Group International Inc. AFX is the company that developed the on-vehicle devices.

## Timing Diagrams

Timing diagrams are used to show the various functions performed by the different components within a technology configuration, and their temporal relationships.

### Timing Diagram When an On-Vehicle Device is Present (GPS Technology Configuration)

The timing diagram depicted in Figure 2 shows the different transactions that take place when a vehicle equipped with a GPS-based on-vehicle device pulls into a gas station and begins a fueling transaction:

1) The DCSS POS system Software sends an XML message to the OSU Software to indicate that a fueling transaction has begun. The XML message from the DCSS POS system Software includes the pump number at which this fueling transaction is taking place.
2) Upon notification that a fueling transaction has begun, the OSU Software sends an XML message to the AFX software to request that vehicle data be retrieved from the device installed in the vehicle located at the pump identified in the XML message.
3) The AFX Software communicates with the on-vehicle device via the Gateway and Fixed Minions and reads the following data:
   - Vehicle Identification Number (VIN)
   - Zone 1-1 mileage data (zone 1 between 0:00 AM to 9:00 AM)
   - Zone 1-2 mileage data (zone 1 between 9:00 AM to 5:00 PM)
   - Zone 1-3 mileage data (zone 1 between 5:00 PM to 0:00 AM)
   - Zone 2 mileage data
   - Zone 3 mileage data
4) The Fixed Minion passes this information to the AFX Software host computer via the Gateway Minion. The AFX Software assembles the information read from the on-vehicle device into an XML file and sends it to the OSU Software.
5) After receiving the XML file, the OSU Software sends an XML message to denote the completion of the transaction to the AFX Software.
6) The OSU Software calculates the VMT fee (after retrieving the prior zone mileage from the central database) using the current VMT rates loaded at startup, and sends this information as an XML file to the DCSS POS system Software.
7) The DCSS POS system Software sends the OSU Software an XML file with *either*:
   - The pump number and fuel purchased for a completed transaction, *or*
   - Notification that the transaction was cancelled (driver opted not to pay VMT fee).
8) For a completed transaction, the OSU Software issues several SQL commands to the Central DB (via modem) to create a new transaction record and update zone mileage data

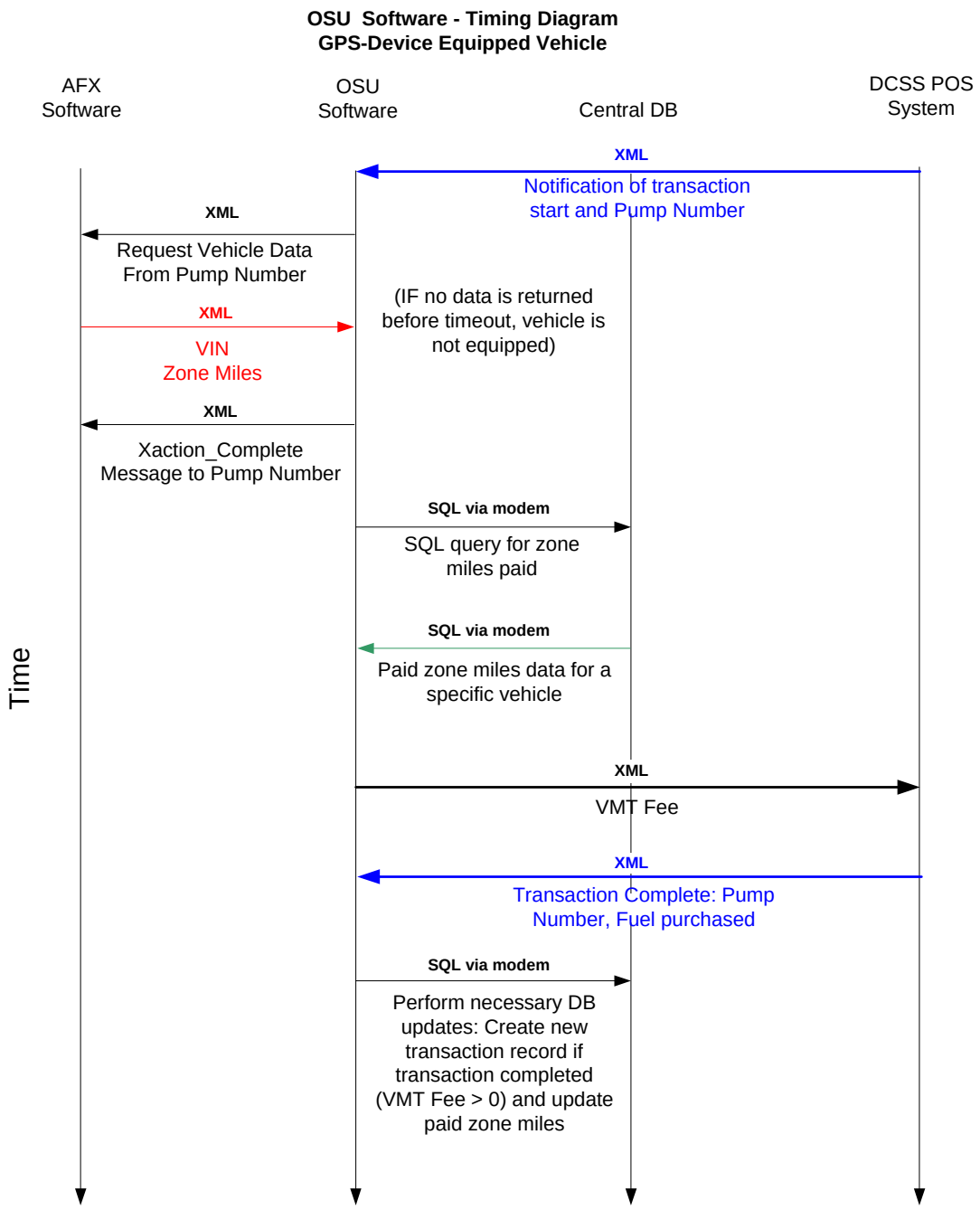


**Figure 2. Timing Diagram When an On-vehicle Device is Present (GPS Technology Configuration).**

**Timing Diagram When an On-Vehicle Device is Present (ODO Technology Configuration)**

Similarly, Figure 3 shows the timing diagram for the different transactions that take place when a vehicle equipped with an ODO-based on-vehicle device pulls into a gas station and begins a fueling transaction:

1) The DCSS POS system Software sends an XML message to the OSU Software to indicate that a fueling transaction has begun. The XML message from the DCSS POS system Software includes the pump number at which this fueling transaction is taking place.
2) Upon notification that a fueling transaction has begun, the OSU Software sends an XML message to the AFX Software to request that vehicle data be retrieved from the device installed in the vehicle located at the pump identified in the XML message.
3) The AFX Software communicates with the on-vehicle device via the Gateway and Fixed Minions and reads the following data:
   - Vehicle Identification Number (VIN)
   - Oregon Miles
   - Non-Oregon Miles
   - Oregon Miles Paid
4) The Fixed Minion passes this information to the AFX Software host computer via the Gateway Minion. The AFX Software assembles the information read from the on-vehicle device into an XML file and sends it to the OSU Software.
5) The OSU Software calculates the VMT fee using the current VMT rates loaded at startup, and sends this information as an XML file to the DCSS POS system Software.
6) The DCSS POS system Software sends the OSU Software an XML file with *either*:
   - The pump number and fuel purchased for a completed transaction, *or*
   - Notification that the transaction was cancelled (driver opted not to pay VMT fee).
7) For a completed transaction:
   - The OSU Software issues several SQL commands to create a new transaction record on the Local DB, *and*
   - Sends two XML files to the AFX Software to update the Oregon Miles Paid mileage information on the on-vehicle device and to denote the completion of the transaction.
8) For a cancelled transaction:
   - The OSU Software sends to the AFX Software an XML message to denote the completion of the transaction.

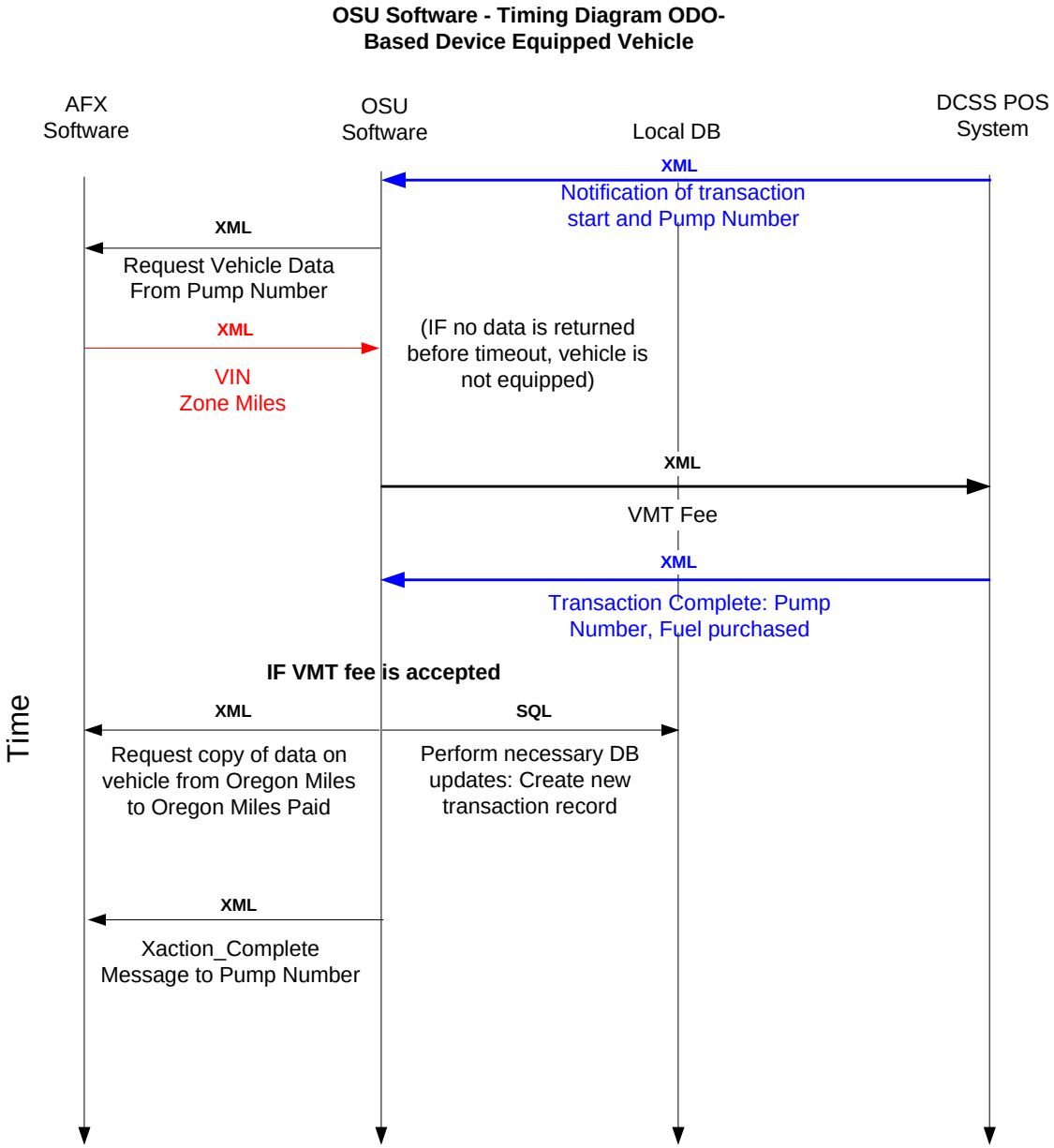


**Figure 3. Timing Diagram when an On-vehicle Device is Present (ODO-based Technology Configuration).**

## Timing Diagram When No On-Vehicle Device is Present

The timing diagram depicted in Figure 4 shows the different transactions that take place when a vehicle not equipped with an on-vehicle device or an on-vehicle device that cannot be read pulls into a gas station and begins a fueling transaction:

1) The DCSS POS system Software sends an XML message to the OSU Software to indicate that a fueling transaction has begun. The XML message from the DCSS POS system Software includes the pump number at which this fueling transaction is taking place.
2) Upon notification that a fueling transaction has begun, the OSU Software sends an XML message to the AFX Software to request that vehicle data be retrieved from the device installed in the vehicle located at the pump identified in the XML message.
3) The AFX Software tries (unsuccessfully) to communicate with an on-vehicle device via the Gateway and Fixed Minions then sends a Xaction_Abort message to the OSU Software.
4) The OSU Software sends to the DCSS POS system Software an XML message with a VMT Fee equal to zero for the vehicle at that pump number.
5) The OSU Software sends a Xaction_Complete message to the AFX Software.

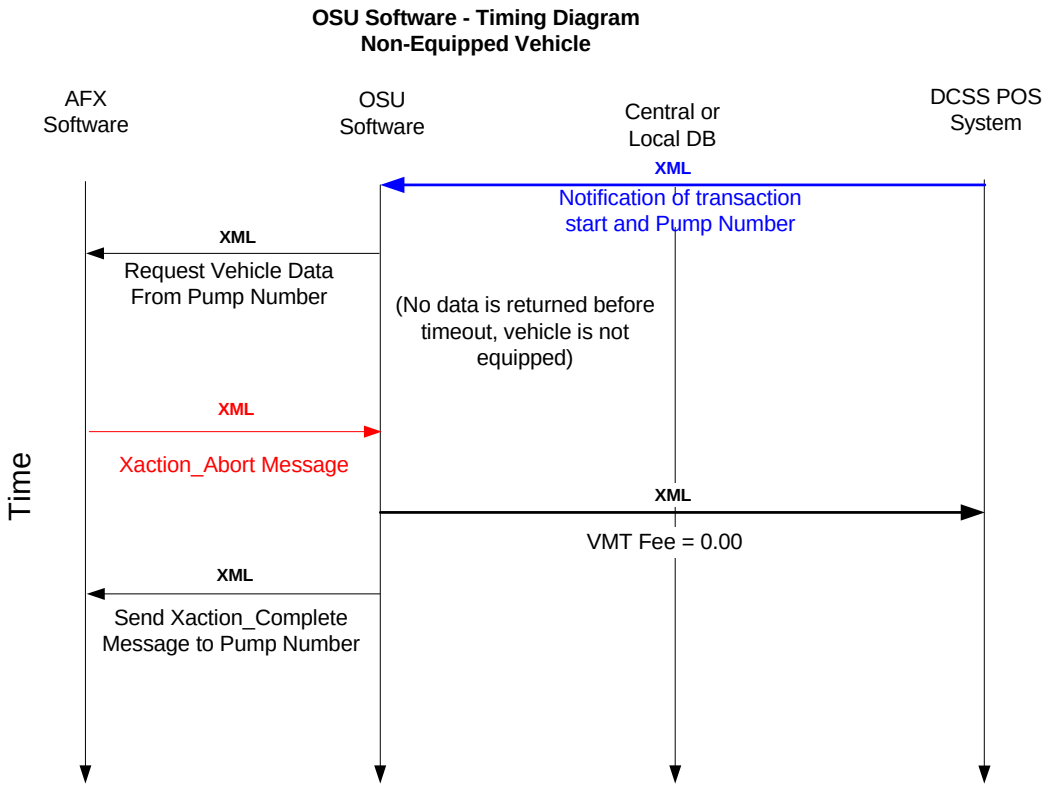


**Figure 4. Timing Diagram When No On-vehicle Device is Present.**

## Platform

The software was developed in Visual Basic (VB) .NET and can be installed on PCs running Windows 98 and above. For testing purposes, the OSU Software will reside on a PC running Windows XP.

## Communication

The OSU Software communicates with the:

- DCSS POS System Software
- AFX Software
- Central DBMS or Local DBMS

Figure 5 illustrates the current setup of the system components for prototype testing of the GPS-based technology configuration. Figure 6 illustrates the system for the ODO-based technology configuration. A more detailed description of both configurations is presented in the following subsections.

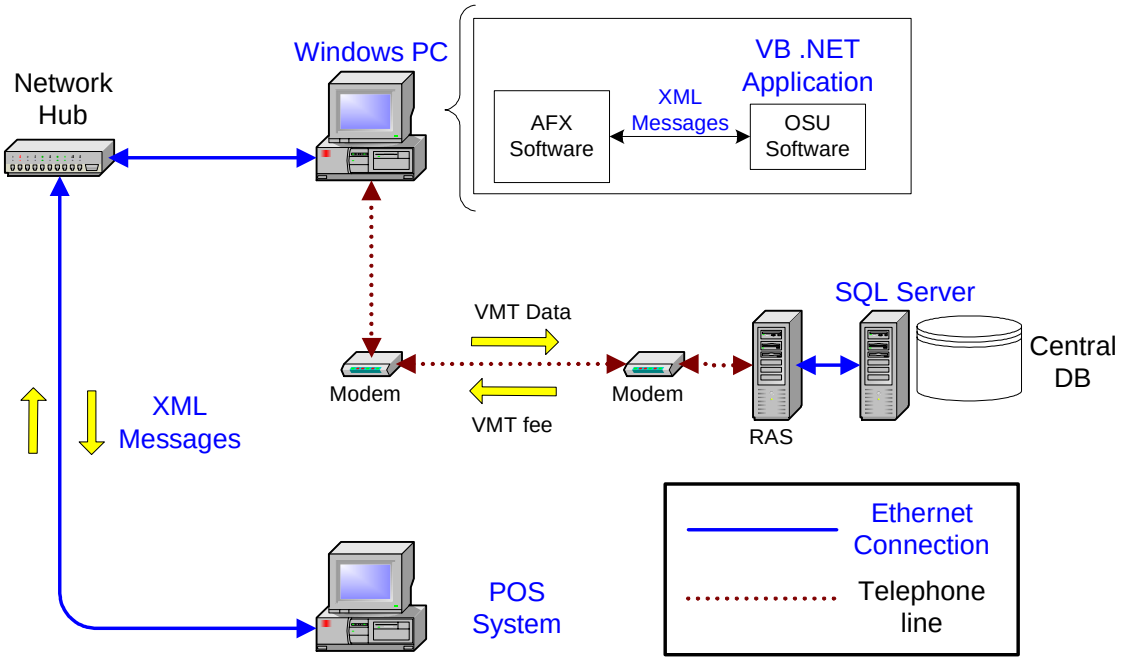


**Figure 5. Physical Setup of System Components for GPS Technology Configuration.**

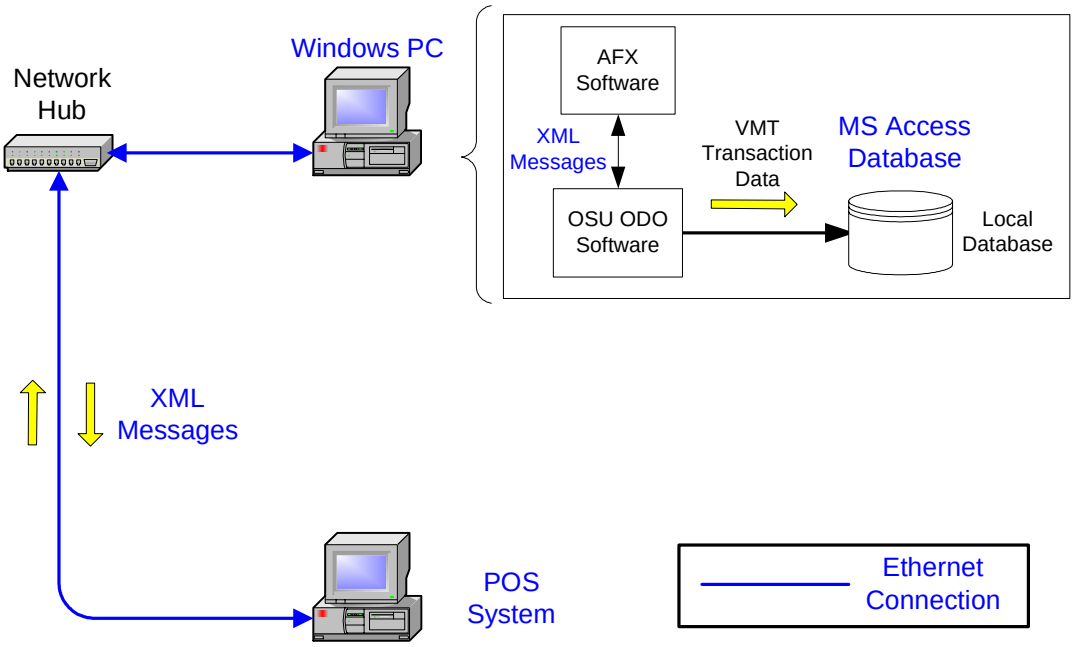


**Figure 6. Physical Setup of System Components for ODO-Based Technology Configuration.**

## DCSS POS System Software

*Physical:* The DCSS POS System Software is located on a notebook computer running Windows XP. The OSU Software establishes a connection with the DCSS POS System Software via Ethernet cable ↔ Network hub ↔ Ethernet cable.
*Protocol:* TCP/IP connection via TCP/IP sockets.
*Data Exchange:* XML files/messages.

## AFX

*Physical:* The AFX Software is located on the PC running Windows XP that also hosts the OSU Software.
*Protocol:* TCP/IP connection via TCP/IP sockets.
*Data Exchange:* XML files/messages.

## Central DB (GPS Technology Configuration Only)

*Physical:* The Central DB is located on an SQL Server running Windows Server 2003 Enterprise Edition. The OSU Software establishes a connection with the Central DB via modem ↔ telephone network ↔ modem ↔ RAS ↔ Ethernet cable ↔ Network hub ↔ Ethernet cable.

> ***Note: The RAS (Remote Access Server) acts as an intermediary, removing the responsibility of managing connections from the SQL Server and adding a layer of security to protect the SQL server.***

*Protocol:* TCP/IP connection via TCP/IP sockets.
*Data Exchange:* Remote stored procedure calls from the OSU Software to SQL Server.

**Local DB (ODO-based Technology Configuration Only)**

*Physical:* The Local DB resides on the same PC running Windows XP as the AFX Software and OSU Software.
*Protocol:* OLEDB Connection.
*Data Exchange:* OLEDB Commands.

## Source Code

Figure 7 depicts a sample of the code included in the OSU Software. Comments and explanations of procedures are in green.

```
'THIS IS THE TCPLISTENER THAT WILL LISTEN TO THE PORT
var_listener = New TcpListener(var_port_num)

'START THE LISTENER
var_listener.Start()

'UPDATE THE STATUS LABEL
Call sub_update_status("Waiting for connection..")

'WAIT HERE UNTIL WE GET A CONNECTION FROM THE OTHER MACHINE
client = var_listener.AcceptTcpClient()

'UPDATE THE STATUS LABEL
Call sub_update_status("Connected to POS system on port 1555")

Dim var_end As Boolean = False

client.ReceiveTimeout = 1000

'CREATE A NETWORKSTREAM
ns = client.GetStream()

Me.cmd_accept_pos_conn.Enabled = False

Me.cmd_disconnect.Enabled = True

Me.PictureBox2.Enabled = True
```

**Figure 7. Sample of OSU Software Code.**

For testing purposes, the OSU Software has a dedicated, synchronous connection with the DCSS POS System Software and the AFX Software. The OSU Software checks once every second for messages originated from the DCSS POS System Software and from the AFX Software. Currently, the system processes a transaction in approximately 2 seconds. Later versions of the software will rely on asynchronous connections, which will reduce transaction time.

## User Interface

The user interface will be designed to allow a user to do the following:

- Configure the OSU Software configuration (i.e. port connection) and IP address of the computer running the AFX Software.
- Start/reboot the server.
- Monitor communication between the OSU Software and various components of the system.

Currently, the user interface is designed to provide communication monitoring.

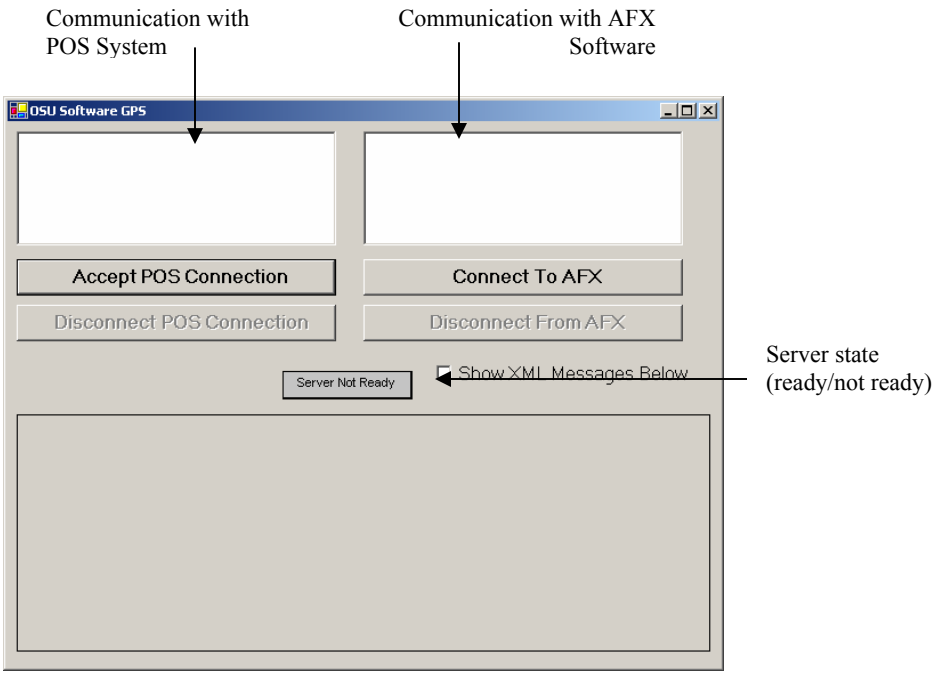


**Figure 8. OSU Software User Interface.**