

Fleet Manager Companion Development Program

The Fleet Manager Companion Development Program, consisting of three software development kits, allows developers to take advantage of open platform architecture of Fleet Manager products. In addition, FM Companion Product developers can leverage their developments off MiX Telematics' world-wide distribution network simply by writing, developing and adapting new or existing solutions to operate with FM Professional software and FM range of on-board computers.

FM Comms SDK

The Fleet Manager Communications (FM Comms) SDK allows a third party device/computer to:

- Get/set a unit's date/time, driver ID, odometer reading, engine hours
- Get general information such as vehicle ID, driver ID, current speed, RPM, odometer reading, GPS data
- Download Event and Interval (Tacho) data from the unit
- Update event configuration and firmware device drivers in the unit
- Store user data and values such as driving and stopping reasons, document codes and costs in the unit's memory

FM Comms SDK can be used by engineers to enable other hardware devices to communicate with the FM Performer on-board computer using RS-232 serial communications.

The SDK includes portable C source code that can be used to implement the FM Performer communications protocol on programmable serial devices, as well as a callable interface that can be used to develop back office applications to communicate with the FM Performer unit once a communications link has been established.

The SDK is written in C and C++ and comes with examples for three operating systems: DOS, Windows and Palm OS.



FM Extension SDK

The Fleet Manager Extension (FM Extension) SDK provides three possibilities for integration:

- Client Extension Manager integration for adding menu items, toolbar buttons, UI elements such as forms, import data into and export data out of FM database, and so much more.
- Server Extension manager component allowing you to read, add, modify or delete FM static object data, gain access to other FM data such as trip/event data, store proprietary data in the FM database, send free-text messages to vehicles, etc.
- XML Web service allowing you to send jobs to vehicles and query the status of jobs in real-time, request the current position and status of vehicles in real-time, and more.

The Fleet Manager Extension (FM Extension) SDK is a collection of documents, tutorials and software tools that allow third party developers to create extensions to the FM Professional software program.

The **FM Client executable** exposes a COM interface known as the Extension Manager interface. Client-side Extensions are add-ons to the Fleet Manager application software. Extensions are COM DLLs which interface with the FM Client executable by implementing a public FMExtension COM interface. Extensions are registered with the Extension Manager object, which loads them at start-up, passing them a handle to itself.

The **FM Server Extension Manager** component is a COM DLL that exposes numerous methods and properties. The components run in-process of the application that instantiates it, thus applications that use this component do not require the FM Client executable.

The **FM XML Web Service** is an industry standard XML web service which runs on a Microsoft Internet Information Service (IIS) web server within the Microsoft .NET framework. The service exposes numerous methods and properties that can be accessed by posting XML requests in SOAP envelopes to the web server using the HTTP or HTTPS protocols.

TERMS AND CONDITIONS APPLY



FM Terminal SDK

The Fleet Manager Terminal (FM Terminal) SDK is a powerful feature that allows for the:

- Complete reconfiguration of menu structure
- Reconfiguration of associated functionality of the FM Terminal

Through the use of menu scripting, the FM Terminal is completely customisable to meet the unique requirements of individual fleets.

Skills needed:

- Basic understanding of scripting and script principles
- An understanding of the concepts of data types – binary, hexadecimal representation
- A great deal of the process involved in writing a script command requires bit and byte level manipulation

The FM Terminal is entirely dependent on the contents of the script and how it will be interpreted. For this reason, a poorly designed script could have a major effect on the operation of the FM Terminal.

DEALER STAMP