
POINTER
Cellocator Division
Pointer Telocation Ltd.
CELLOCATOR
Integration workshop
Cello-IQ – Driving Intelligence Delivered. January, 2013.

Objectives

By the end of this lesson you will be able to:

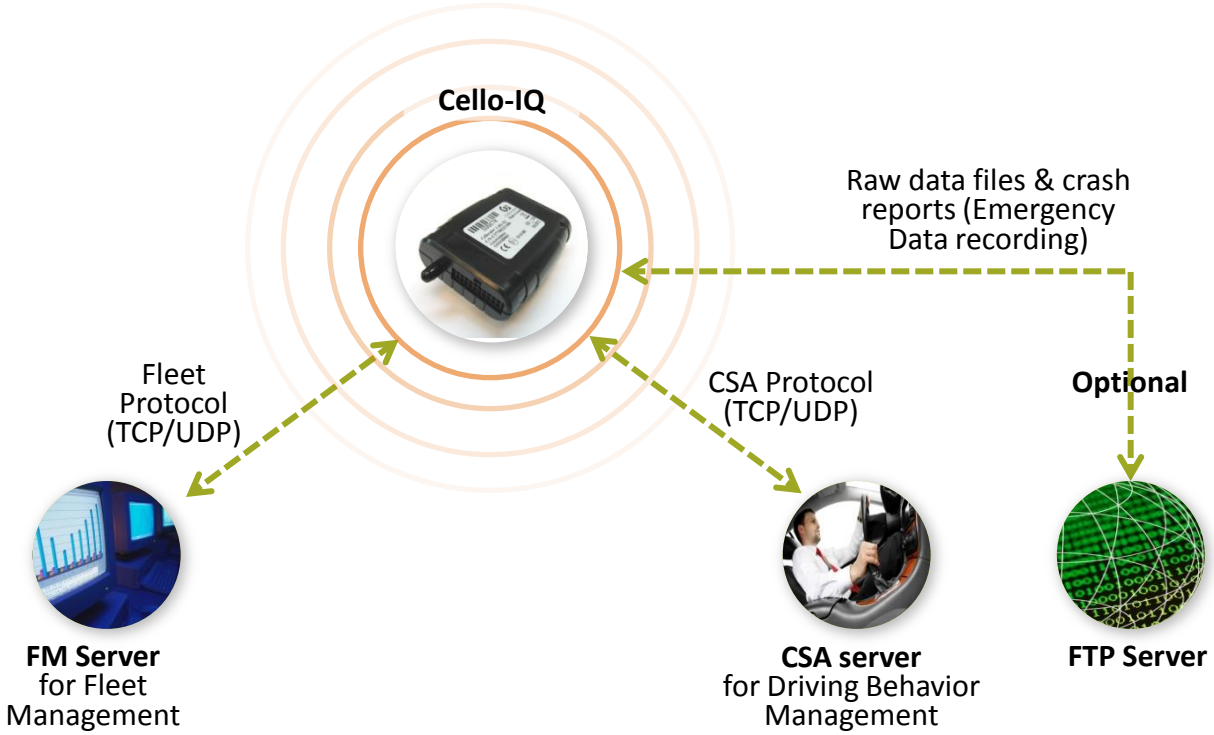
- Understand the backend network architecture for integrating Cello-IQ
- Get familiar with the 3 integration option with Cello-IQ
- Understand the Pros and Cons of each option
- Be familiar with the integration processes

Topics

- **Introduction**
- Introducing the 3 integration options for Cello-IQ
  - Integration of 3rd party DBM to an existing FM platform
  - DBM integration package
  - DBM Self implementation

Introduction to Cello-IQ integration

- Cello-IQ integrates driver's **Safety and Eco Driving** analysis (DBM) into the legacy **Fleet Management** core, using multiple concurrent IP sessions

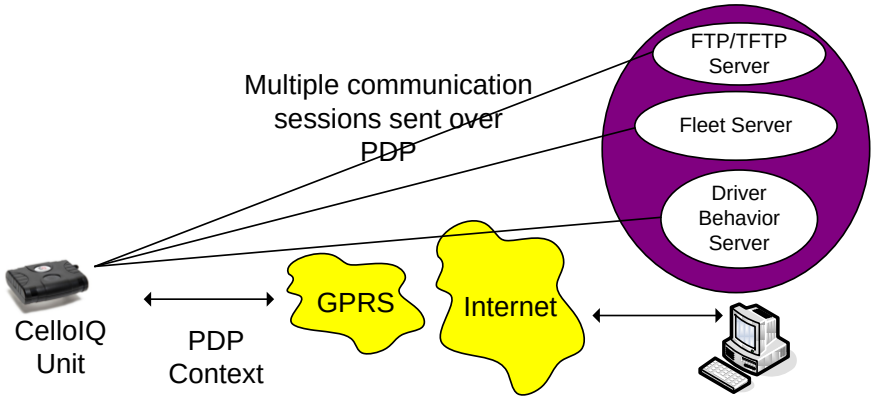


- Cello-IQ communicates with the **FM server** (Fleet Management application), the **CSA server** (for CSA data management) and optionally, the **FTP/TFTP server** (for CSA raw data files and crash reports [Emergency Data recording] transfer)

Network Architecture

- **Multiple concurrent IP sessions network architecture**

- Unlike fleet management units using single session, Cello-IQ supports multiple sessions, separating **Fleet Management** data interchange and **CSA** data interchange
- It also support **File Transfer Protocol** carrying raw data files of trips, maneuver statistics and crash data to the **FTP/TFTP server**
- The **Fleet Management** protocol has higher priority over the **CSA protocol** since it handles the basic APN session maintenance and control
- CSA offers new **configuration fields** designed to set the new session properties (IP/DNS/port) and the FTP/TFTP client side properties



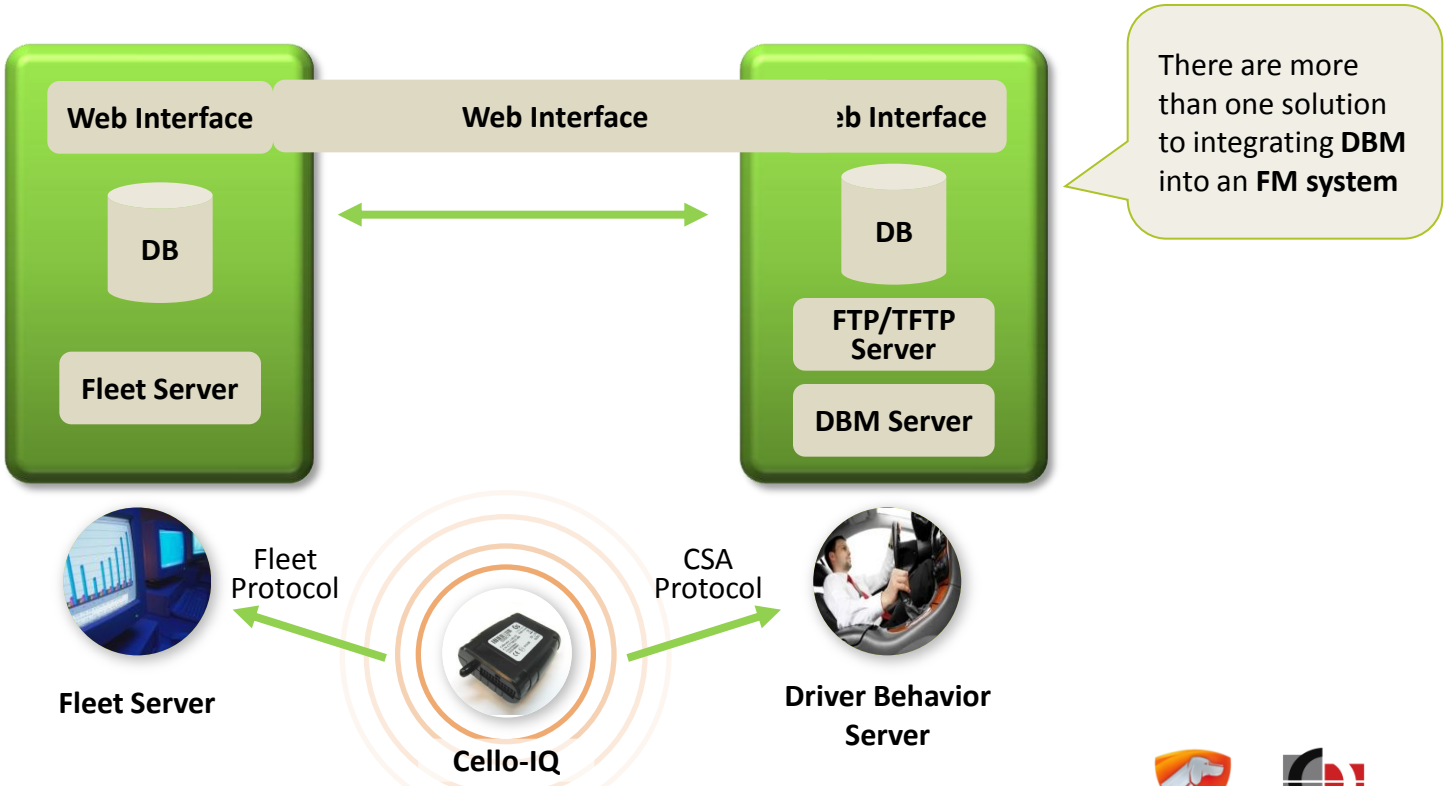
POINTER

Cellocator Division
Pointer Telocation Ltd.

Network Architecture

Cello-IQ communication architecture

- The diagram describes the flow of information between **Cello-IQ** and the **FM / DBM** servers
- Each of the servers receive different data from the unit and stores, process and finally present it on a Web interface



Network Architecture

- **Network Architecture advantages**

Great Flexibility

Multiple socket architecture provides great flexibility when building the back office infrastructure and offloads application layer

DBM independent of FM

Has the possibility to work with distinct SW modules, in charge of the Driver Behavior Management application, independently of the Fleet Management application

Seamless Integration

Provides seamless integration of a 3rd party's DBM application into TSP's or SW integrator's fleet management platform

Multiple Socket Architecture

Topics

- Introduction
- **Introducing the 3 integration options for Cello-IQ**
  - **Integration of 3rd party DBM to an existing FM platform**
  - DBM Integration Package
  - DBM Self Implementation

Cello-IQ integration options

- **There are 3 options of integrating Cello-IQ with a back office SW**

Integration of 3rd party DBM application

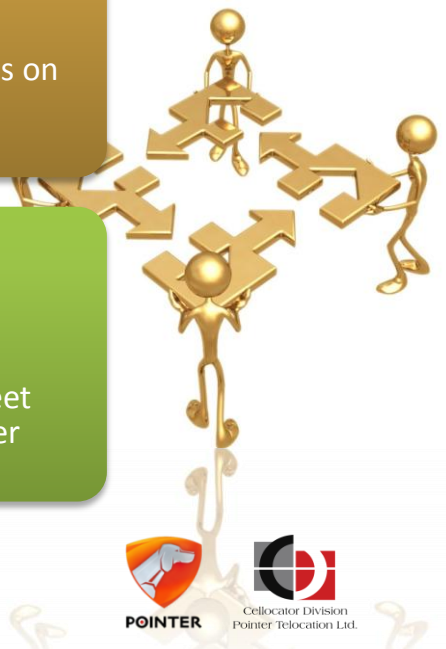
- Integration of 3rd party DBM application to an existing FM platform
- Provides a quick integration process with a fully developed solution
- Targeted for customers still considering their options, willing to avoid self development or anxious to reach short TTM

Using Cellocator Integration package

- Connect full Cellocator - DBM integration package to an existing FM platform
- For customers whishing to focus on the application layer itself and less on the connectivity and GW layers

DBM Self implementation

- DBM Self implementation as add-on to an existing FM platform
- Providing a full control on all the low level communication layers
- Targeted at new customers and those experienced with Cellocator Fleet management and wishing to extend their capabilities to support Driver Behavior Management



Integration of 3rd party DBM

- Cello-IQ is ready to integrate with a 3rd party DBM application

Ready solution

Cello-IQ can be used as an end-to-end solution when integrated with an off-the-shelf proven 3rd party DBM application

Short TTM

Allowing TSP's to deliver in a very short TTM (time to market) Driving Management capabilities

For New to Cello-IQ

Intended for TSPs and SW integrators wishing to extend their capabilities to support driver behavior and are new to Cellocator's APIs and OTA Interfaces

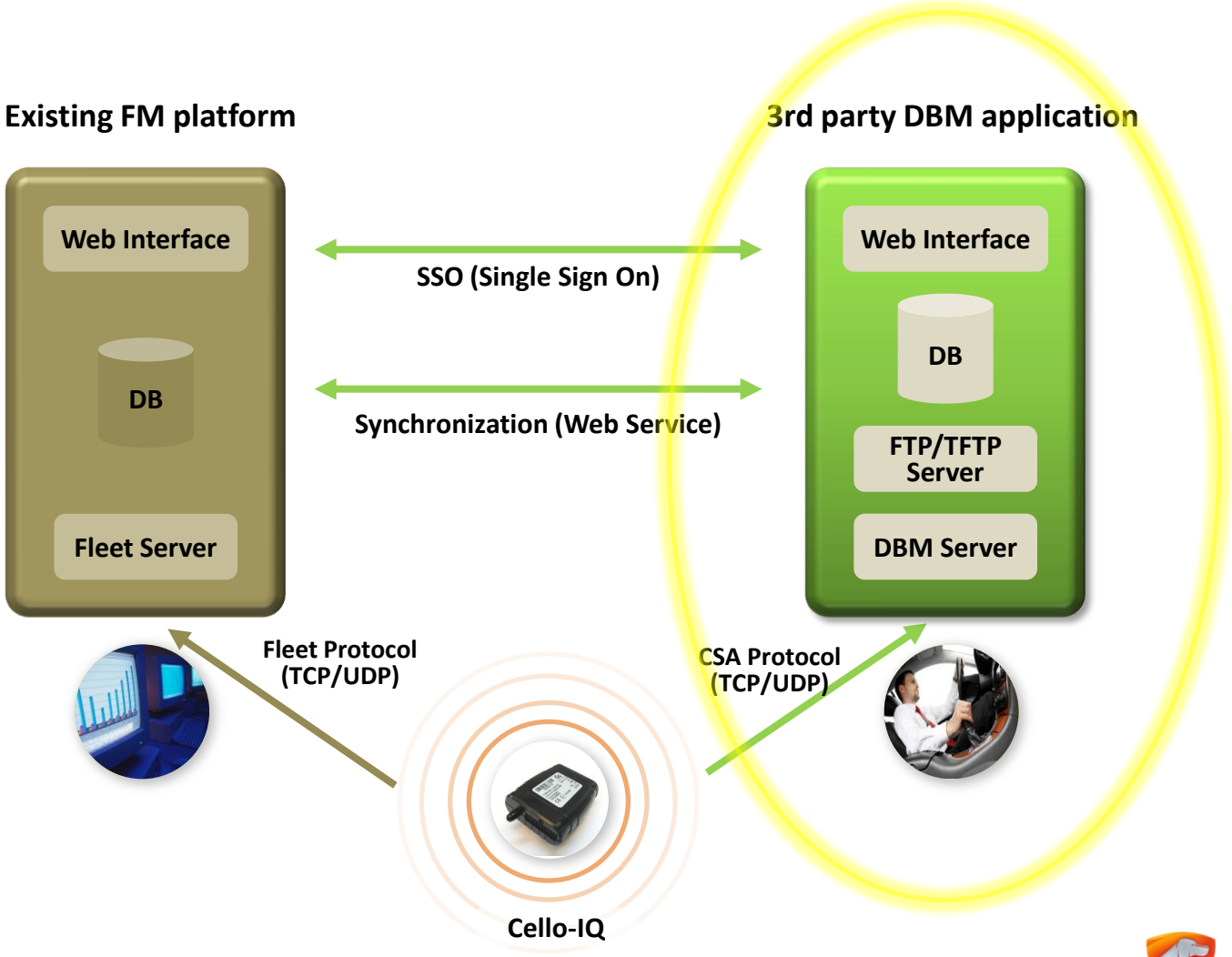
3rd Party DBM Application Solution

POINTER

Cellocator Division
Pointer Telocation Ltd.

Integration Architecture

- Integration of 3rd party DBM application to an existing FM platform – diagram



Integrating with a 3rd Party Advantages

- Customers integrating with a 3rd party DBM application are benefitting from:

✓ Quick and easy integration process

✓ Cost reduction in development hours

✓ Quick Time to Market

✓ Avoided low level management of data bases, FTP servers etc.

✓ Scalable solution, allowing to develop in stages according to one needs

✓ Providing DBM capabilities quickly, and in the mean time allowing continuing development at a later stage

Typical Off-the-shelf DBM app Features

Analytics Dashboard

- Dashboard configuration widgets
- Fleet view – 30k feet
- Filter & search fleet and/or drivers
- Filter time periods
- Comparisons and trends
- Tabular / Graphical performance representation
- Table customization
- Reporting

Insights

- Drill-down with details on a vehicle or driver data
- Widgets configuration for detailed vehicle or driver dashboard
- Filter time periods for the detailed dashboard
- Trip level Safety & Eco scores
- Trip story as list / on map including events and maneuvers
- Safety and Eco events comparison charts between vehicle/driver and fleet / larger population
- Reporting
- Driver score card

Full function Drill Down

- Detailed Safety & Eco events with time-stamp, geographical and dynamics information
- Event classification and severity details
- Full reporting solution including event / maneuver / collision playback
- Web services availability for back office application
- EDR analysis module. detailed information for accident restructuring and investigation

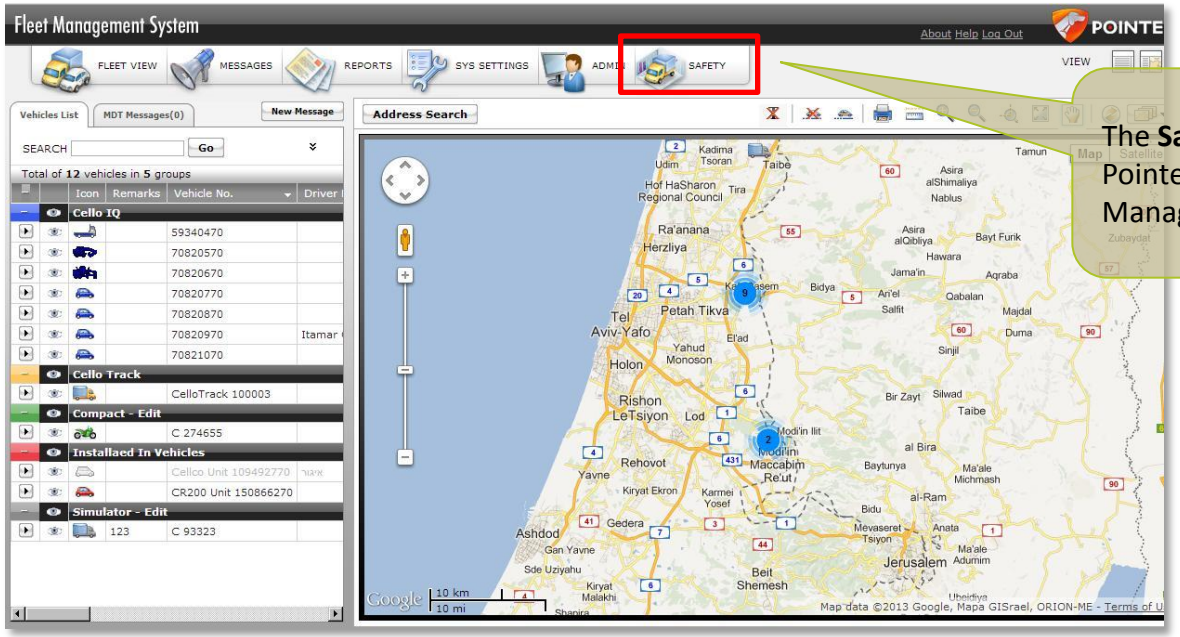
Integration Steps

- **When integrating 3rd party DBM application to your existing FM platform, you need to:**
  - Configure your system's **APN** to allow communication with the DBM application
  - Follow application providers provider's instructions regarding other **infrastructure** required and prerequisites.
  - The integration process includes:
    - **GUI** implementation
    - **SSO** (Single Sign On) mechanism definition
    - **Bi Directional** data synchronization

GUI Integration

- Create a reference tab to the **DBM application** landing page

- From your **Fleet Management** platform, create a reference such as a '**Safety**' tab, to direct the user to the **DBM application** landing page



The **Safety** Tab in Pointer's fleet Management System.

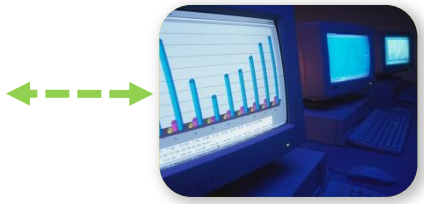
POINTER

Cellocator Division
Pointer Telocation Ltd.

SSO (Single Sign On)

- **Single Sign On** mechanism – login once, move between 2 web applications without the need to login to the second system

Fleet Management portal's user



DBM web application

Login once

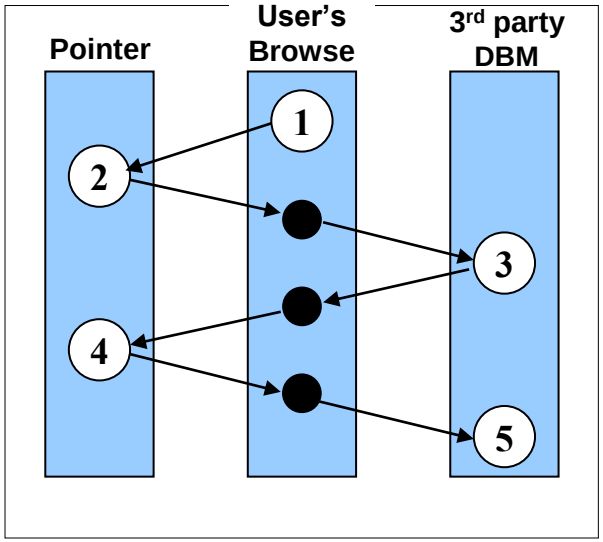
- ✓ Single login action in the **FM** web site
- ✓ Secured authentication and login in to the **DBM** web application, with no additional login action
- ✓ Simplifies user navigation
- ✓ Enhances the web sites usability

SSO Web service Flow

- **The diagram below describe the flow of SSO process**

- The 3rd party DBM application will provide a dedicated login page (e.g. SSOTransfer.aspx) that handles the SSO login process

- The customer's **FM** portal requests this page (with HTTP GET) supplying encrypted query strings



SSO Login Flow

- **The secure login process include the following:**

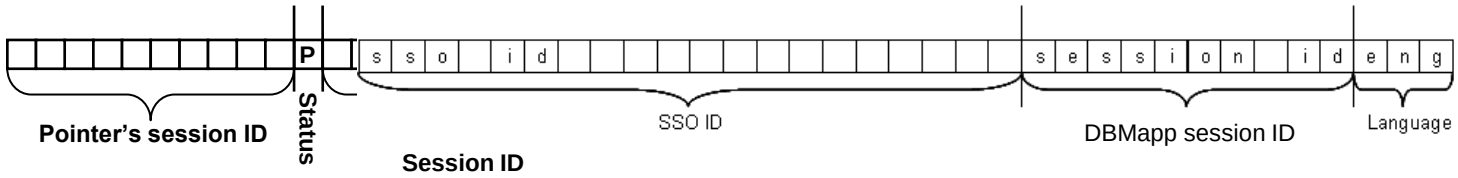
1. The user attempts to reach the **3rd party DBM application** web site by clicking the **DBMapp** link on **FM**'s web application.

2. **Pointer** generates a 10 digit unique id for the user's browser session and send it as an encrypted request to **DBMapp**.

3. **DBMapp** site retrieves the query string's "login", decodes, decrypts and creates a new id query string value.

4. **Pointer** retrieves the id query string, decodes, decrypts and verify it. Then a second request message for a SSO login is made.

5. **DBMapp** decrypts the query string, validates and match the session id against the current session id. Upon success, username / password are retrieved and user is signed on to the **DBMapp.**



POINTER

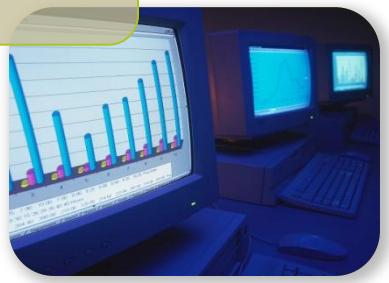
Cellocator Division
Pointer Telocation Ltd.

Bi Directional data synchronization

- The **FM** and **DBM** systems should be synchronized, to make sure all changes and updates made on the FM system as part of provisioning and daily account management are reflected in DBMapp portal
- The integrator will implement a Set of Functions to allow for Slave/Master relations between the **FM** and **DBMapp**
- **Bi directional** data synch is needed to ensure data comparison and integrity

Fleet Management

Data synchronization with **DBMapp**. The relevant web services are invoked by **the FM app**.



DBMapp web application

Changes and updates such as a new driver added, account or vehicle update etc.

Each system should have the ability to run comparison on the data structure of the other system, to insure data integrity.

POINTER

Cellocator Division
Pointer Telocation Ltd.

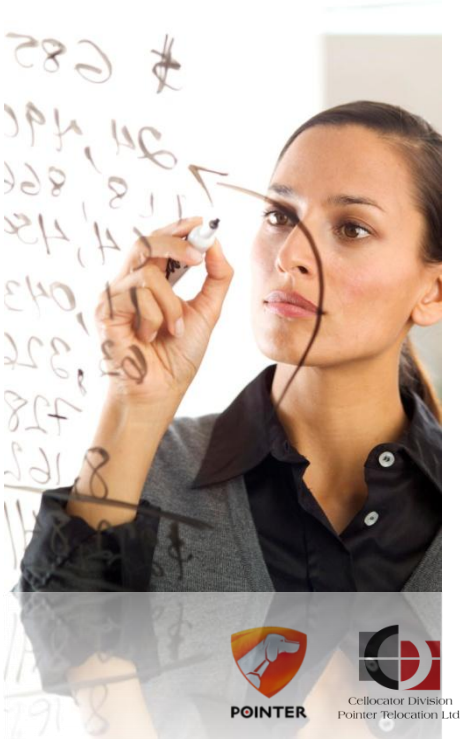
Additional Information

- You can also find more information on integrating with a 3rd party DBM at:

- **DEMO site for SSO integration:**

  http://212.143.168.169/fleet/

  To enter this site please contact **Cellocator Support Team** for credentials.
  sales@pointer.com, Tel: +972-3-572311, Fax: +972-3-5723100



Topics

- Introduction
- **Introducing the 3 integration options for Cello-IQ**
  - Integration of 3rd party DBM to an existing FM platform
  - **DBM Integration Package**
  - DBM Self Implementation

Introduction to DBM integration package

- **What is Cellocator's Integration package?**

Full GW Solution

Consists of Cellocator GW (**Communication Server**, **Parser)** and **Database API** in a modular, scalable, balanced and redundant structure which can run on any standard MS platform.

For New to Cello-IQ

Intended for TSPs and SW integrators using Cellocator's integration package or those who are new to Cellocator's APIs and OTA Interfaces.

Simplified Solution

A set of **Backend Gateway Components** designed to transform **low level** communication & protocol management task, into a **database level**, explicit and parsed information integration mission.

Easy to implement

Integrators using the **Integration Package** will implement data-base level integration between their application layer and the units represented as data-base entries.

Free of charge **Cellocator Integration Package**

Introduction to DBM integration package

- **Customers using Cellocator Gateway are benefitting from:**

✓ Quicker and easier integration process

✓ Entitled for software upgrades, technical support and more

✓ A local and safe system

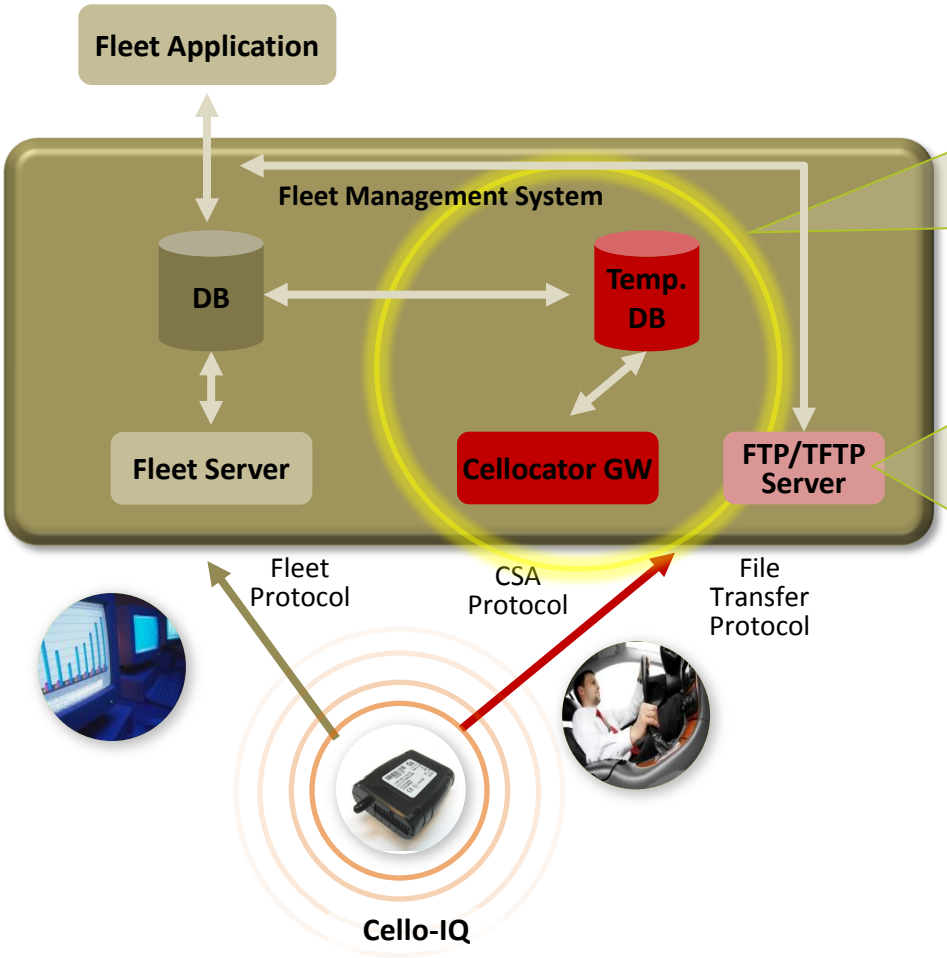
✓ Full control on different system components such as servers used etc.

✓ Scalable system, allowing to develop in stages according to needs

✓ Providing DBM capabilities quickly, while continuing development later on

Integration Architecture

- **DBM Integration package with an existing FM platform – diagram**



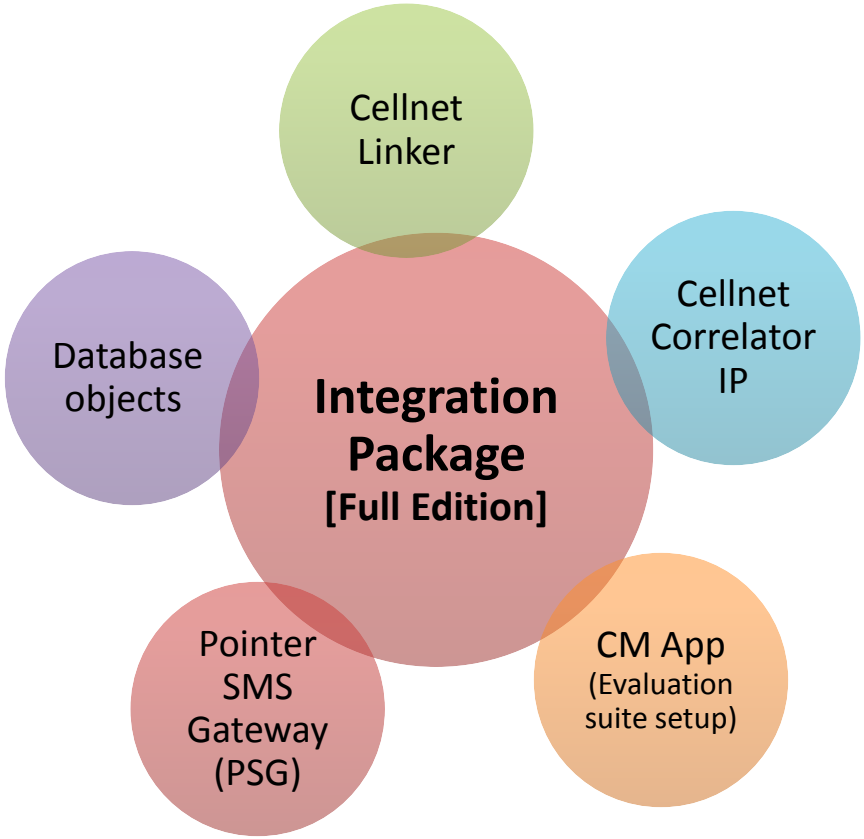
The **Integration Package** provides customers wishing to integrate Cello-IQ with their Fleet Management environment, a complete and quick to implement solution.

Raw data files sent by Cello-IQ are received via File Transfer protocols and saved by the FTP server added to the server side solution.

The server side application should be able to access these files and parse them for more detailed information and analysis.

Package Contents

- **The Integration Package (full edition) contains:**



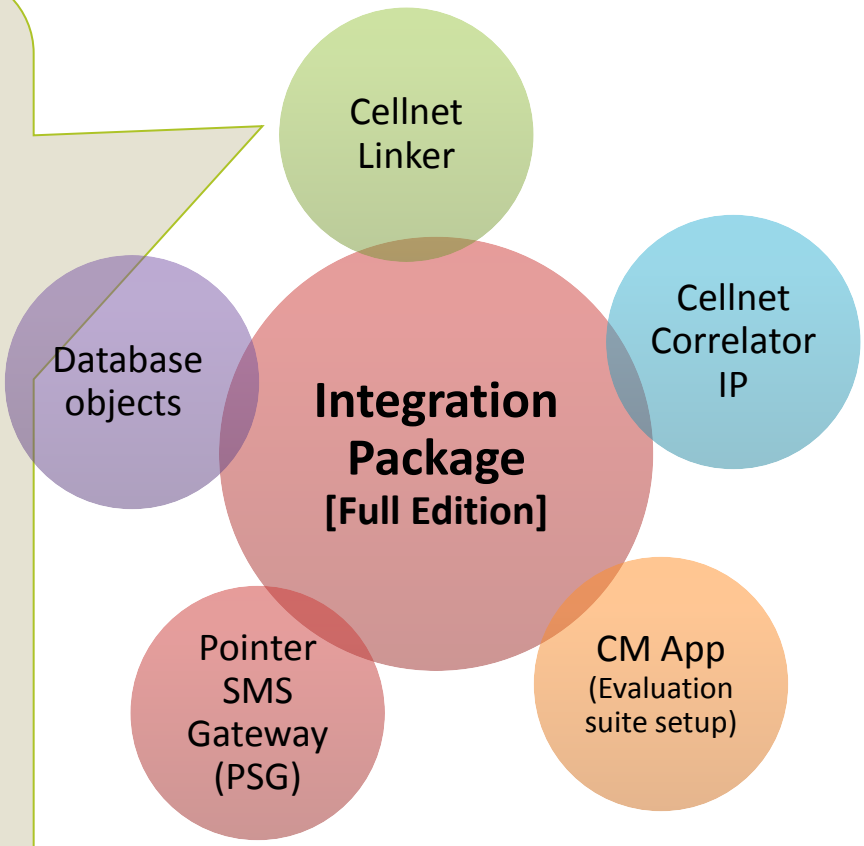
Package Contents

- **The Integration Package (full edition) contains:**

Cellocator's Cellnet Linker application - supports MCGP OTA protocol. 1st tier GPRS communication application written in C# .Net platform.

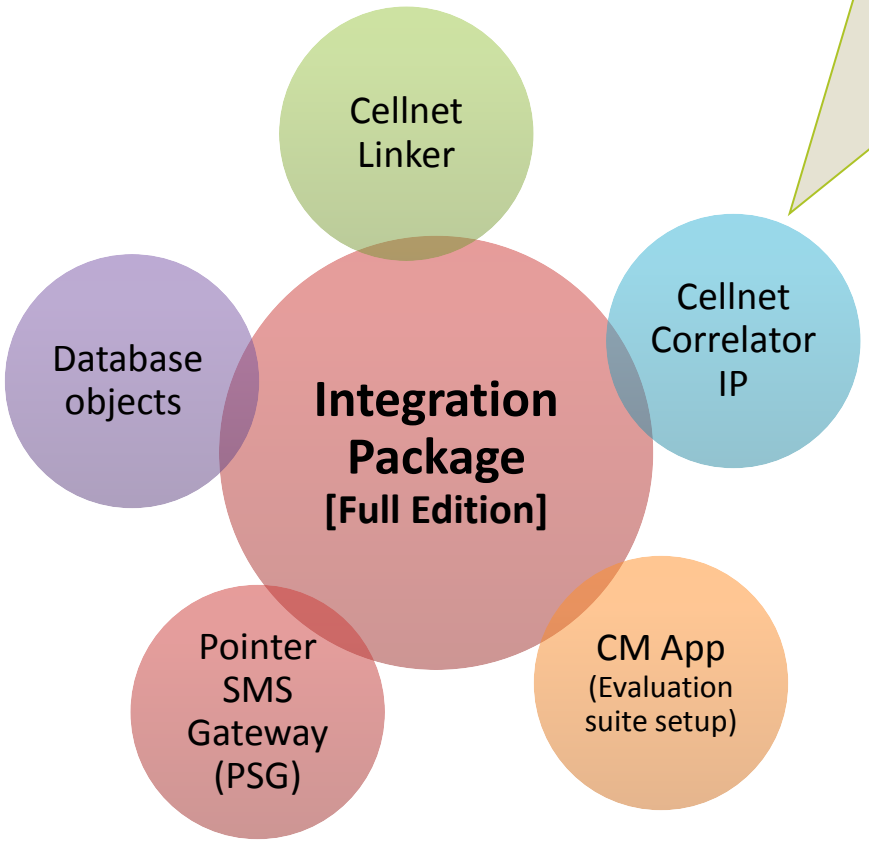
It communicates directly by sending downlinks commands and processing uplinks messages from Cellocator units while communicating with the **Cellnet Correlator** application using **Microsoft Messaging service**.

The **Cellnet Linker** assimilates the **GPRSManager.dll** file as a set of functions and APIs which enables bi-directional communication with Cellocator units, including IP/Port/Socket management, monitoring and other management features.



Package Contents

- **The Integration Package (full edition) contains:**

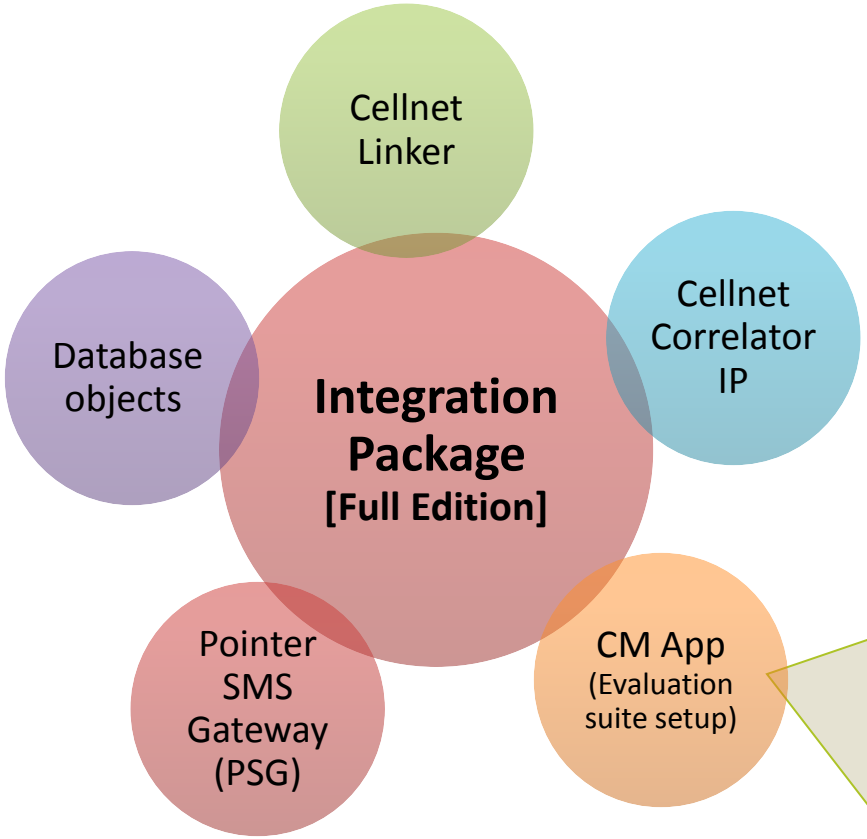


Cellocator's Cellnet Correlator application is a 2nd tier GPRS$ SMS communication application written in C# .Net platform.

It communicates with the Cellnet Linker application for MCGP/S protocol using Microsoft Messaging service (MSMQ) and CM App for CSA protocol via Pipe mechanism by using CC App dll, for sending downlink messages and receiving uplinks messages from Cellocator units, while communicating with Microsoft SQL Server where it stores parsed uplink messages and retrieves downlink messages designated for Cellocator units.

Package Contents

- **The Integration Package (full edition) contains:**



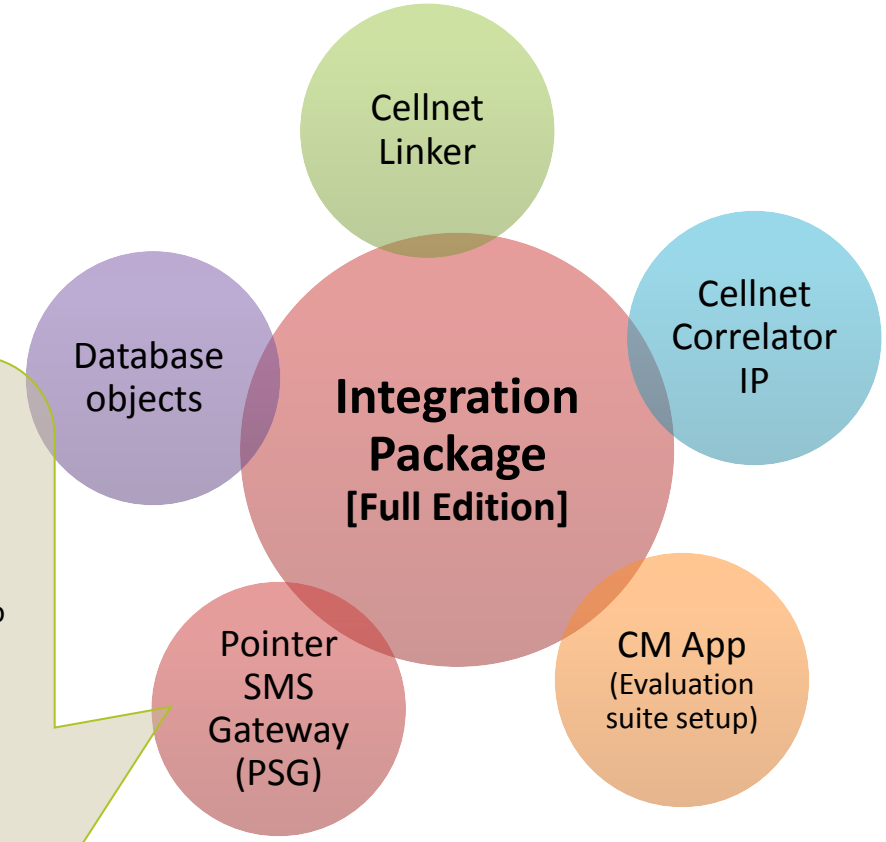
Cellocator's CM Application – supports CSA OTA protocol. CM App is a 1st tier GPRS communication application written in C# .Net platform.

It communicates directly by sending downlinks commands and processing uplinks messages, CSA related from Cellocator units while communicating with the Cellnet Correlator application using Pipe service.

Both **Linker** and **CM App** can work simultaneously with the same devices on different ports on the same server.

Package Contents

- **The Integration Package (full edition) contains:**



Pointer SMS Gateway (PSG) application is a 1st tier SMS communication application written in ASP.Net platform.

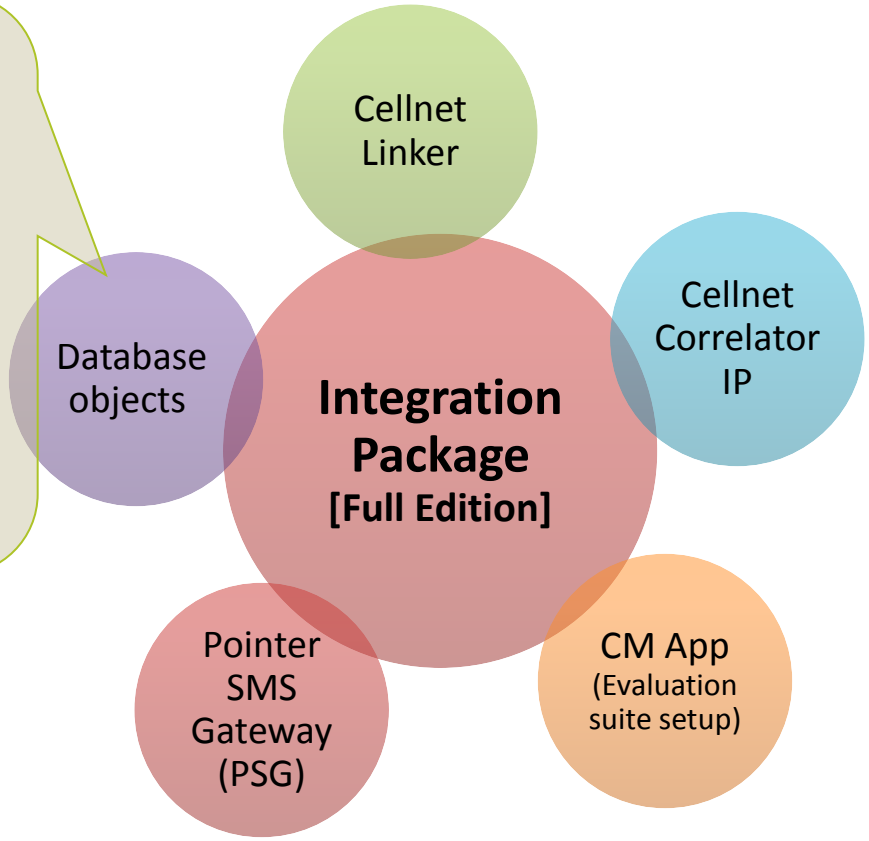
It communicates directly (PSG uses a 3rd party application to send/receive SMS by sending downlinks commands and processing uplinks messages from Cellocator units while communicating with the Cellnet Correlator application using **Microsoft Messaging service** (uplinks) and **HTTP requests** (downlinks).

Package Contents

- **The Integration Package (full edition) contains:**

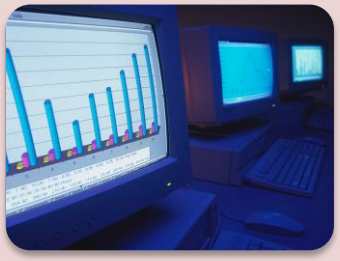
Cellocator utilizes **Microsoft SQL Server** technology in order to communicate with the **Cellnet Correlator** application. This is done via a set of database tables and stored procedures.

The developer can also build their own set of tables for Cellocator uplinks and downlinks and utilize the store procedures accordingly in order to pull and place uplinks and downlinks.



Integration Package Compatibility

- **Integration Package can be used with most of Microsoft Windows operating systems & SQL Server platforms with the following limitations:**



Cellnet Linker, Cellnet Correlator & CM App

- Microsoft Windows Server 2000 family and above. 64 bit compliance (32 bit application)

Pointer SMS Gateway (PSG)

- Microsoft Windows Server 2003 family and above. Both 32bit and 64bit environments are supported

Database Object

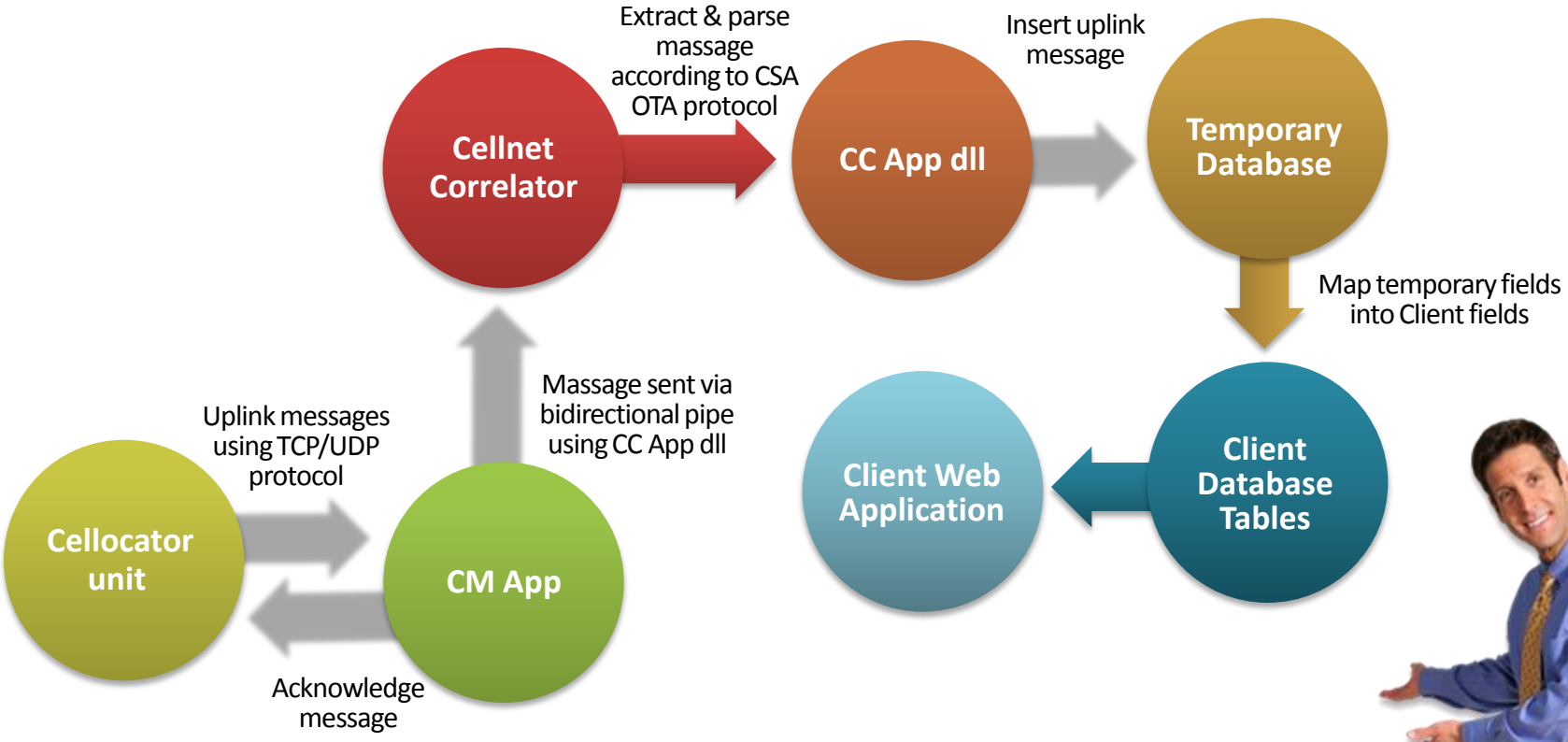
- Microsoft SQL database starting from SQL Server 2000 up to SQL Server 2008, Express, Standard or Enterprise editions. Both 32bit and 64bit environments are supported

- Instance default language MUST be English

Cellocator GW Integration

- Install all the components of the **Integration Package full Edition**. Now we can view how the Cellocator DBM and FM protocols are mapped to the existing FM system.

Uplink Message Flow - CSA protocol



POINTER

Cellocator Division
Pointer Telocation Ltd.

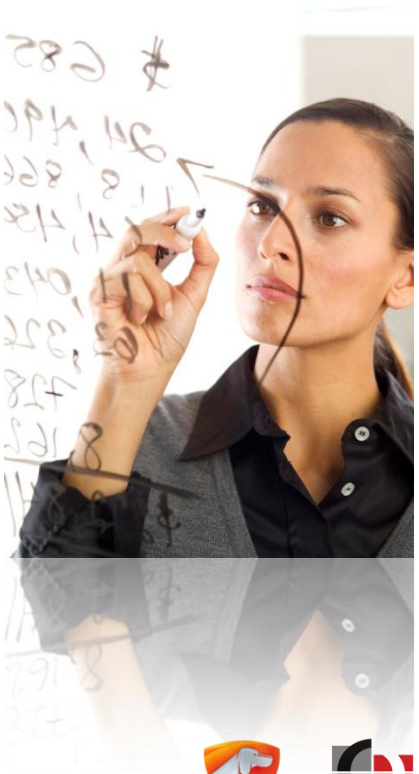
Integration Package Integration

- **Uplink Message Mapped data - examples**

Field name	Field description	Type	Example
MsgProtocol	OTA protocol type: MCGP = 0 CSA = 1	Small integer[2 bytes]	0,1
TripId	Trip Id in CSA protocol	Integer	521123
ManeuverId	Maneuver Id in CSA protocol	Integer	521123

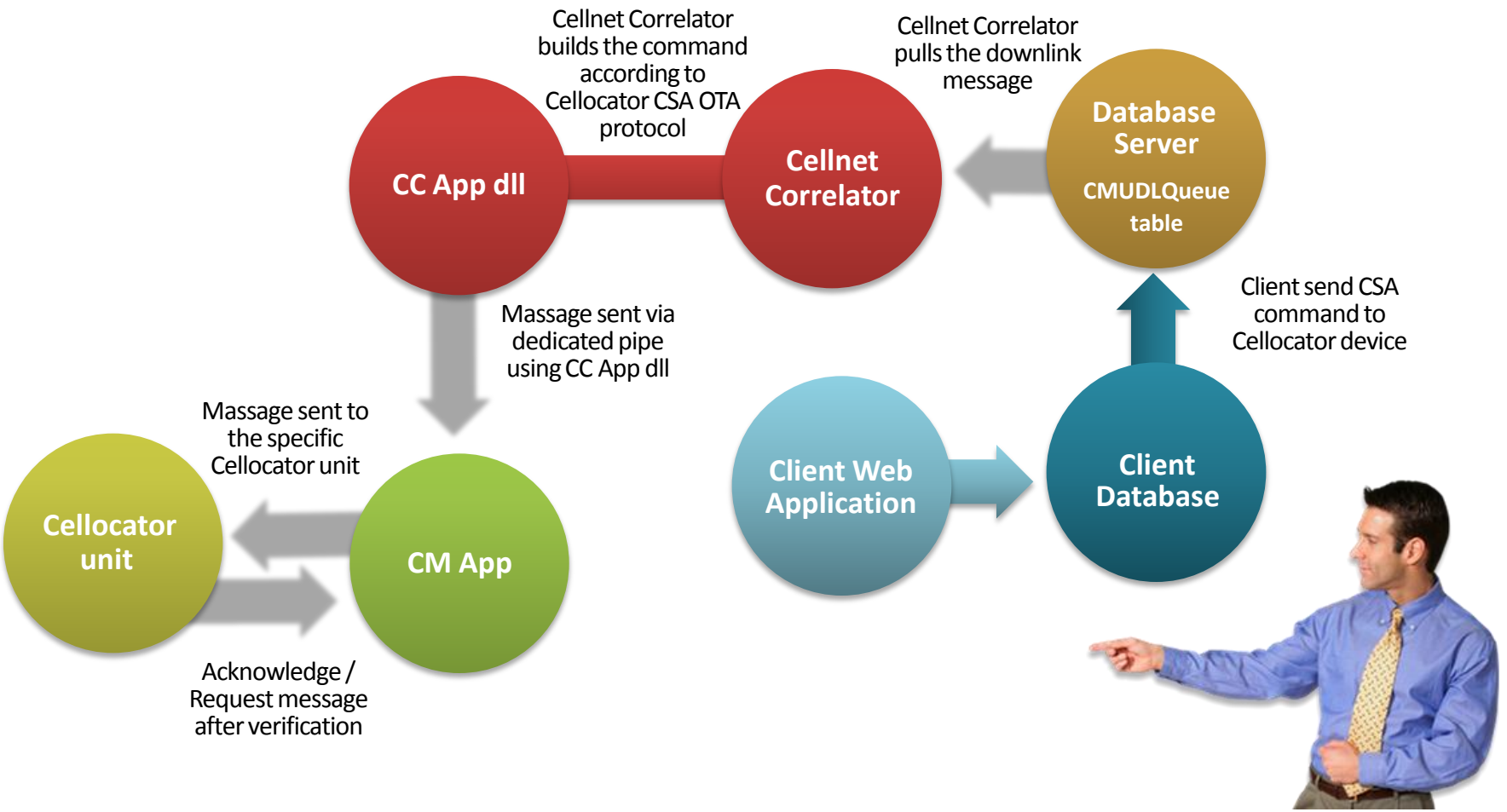
- **Module 30 parsing example**

Field name	Data	Parsing
EventReason	7	Hard Braking
EventSubReason	1	Green Severity
…		
OperationMode	1	Engine On, Calibrated
…		
TripID	556	Sequential numerator
ManeuverID	9	Sequential numerator
…		
AccidentBufferStatusBitmak	0	Buffer empty
HDOP	4	High quality Fix
SatellitesUsed	10	Integer
…		



Cellocator GW Integration

Downlink Message Flow - CSA protocol



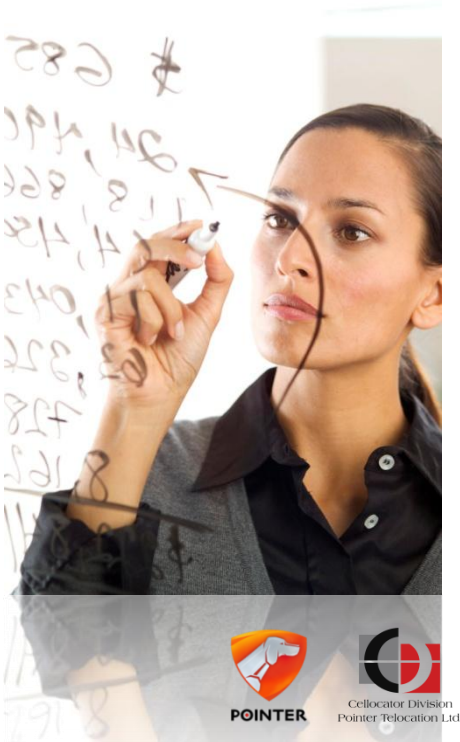
Integration Package Integration

- You will find more information on Integration Package in the **Cellocator Integration Package [Full Edition]** manual

- Now you can proceed to develop the **GUI** within your existing **Fleet Management** platform according to specific needs

Additional Information

- You can also find more information on the DBM integration package at:
  - Cellocator's integration tools web page: http://www.cellocator.com/knowledgebase/integration-tools/

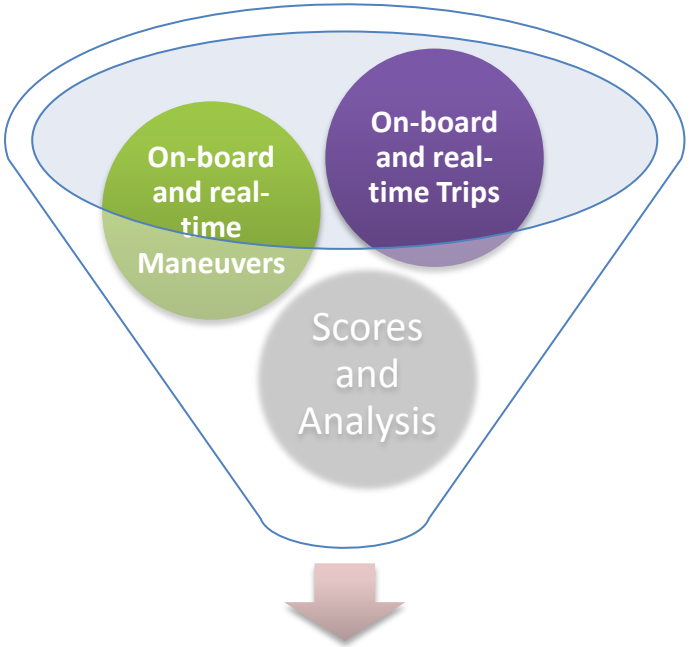


Topics

- Introduction
- **Introducing the 3 integration options for Cello-IQ**
  - Integration of 3rd party DBM to an existing FM platform
  - DBM Integration Package
  - **DBM Self Implementation**

Introduction to DBM Self Implementation

- **Cello-IQ is ready for integration with any TSP's SW platform with relatively minimal development and integration effort.**
- Cello-IQ analyzes and scores, on-board and in real-time, all maneuvers and trips.



✓ The only mission left is to build a driver management portal and Device communication management layer.

POINTER

Cellocator Division
Pointer Telocation Ltd.

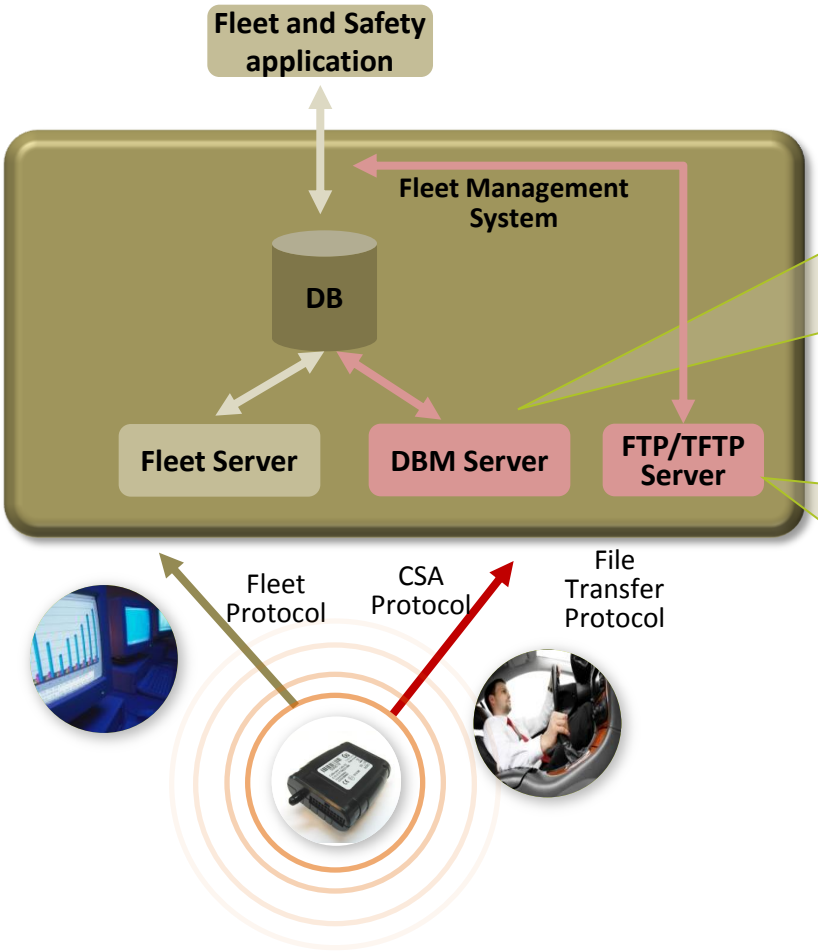
Introduction to DBM Self Implementation

Self Implementation process is targeted at 3 types of audiences:

- Developers adding Cello-IQ units to their existing Cellocator Fleet Management system, familiar with Cellocator MCGP protocol, self-implementing all the low level communication layers
- Integrators modifying their server to support CSA protocol in addition to an existing Fleet Management protocol
- Integrators developing their own Fleet Management and Cellocator DBM solutions

Self Implemented Integration Architecture

- **DBM Self Implementation to an existing FM platform – diagram**



The **CSA server** side is developed and added to an already implemented and running **FM server**.

Fleet sessions have higher priority over CSA sessions. It is mandatory to have an open session with the Fleet server prior to the establishment of CSA server.

Raw data files sent by Cello-IQ are received via File Transfer protocols and saved by the FTP server added to the server side solution.

The server side application should be able to access these files and parse them for more detailed information and analysis.

Self Implementation Advantages

- **Customers self-implementing Cello-IQ are benefitting from:**

✓ Full control over solution features and parameters

✓ “Tailored made” solution specific to one’s needs

✓ Independent from any outside solutions or 3rd party components

✓ Open codes and protocols which can be updated and changed according to specifications

✓ An option to develop a unique solution

Cellocator Integration

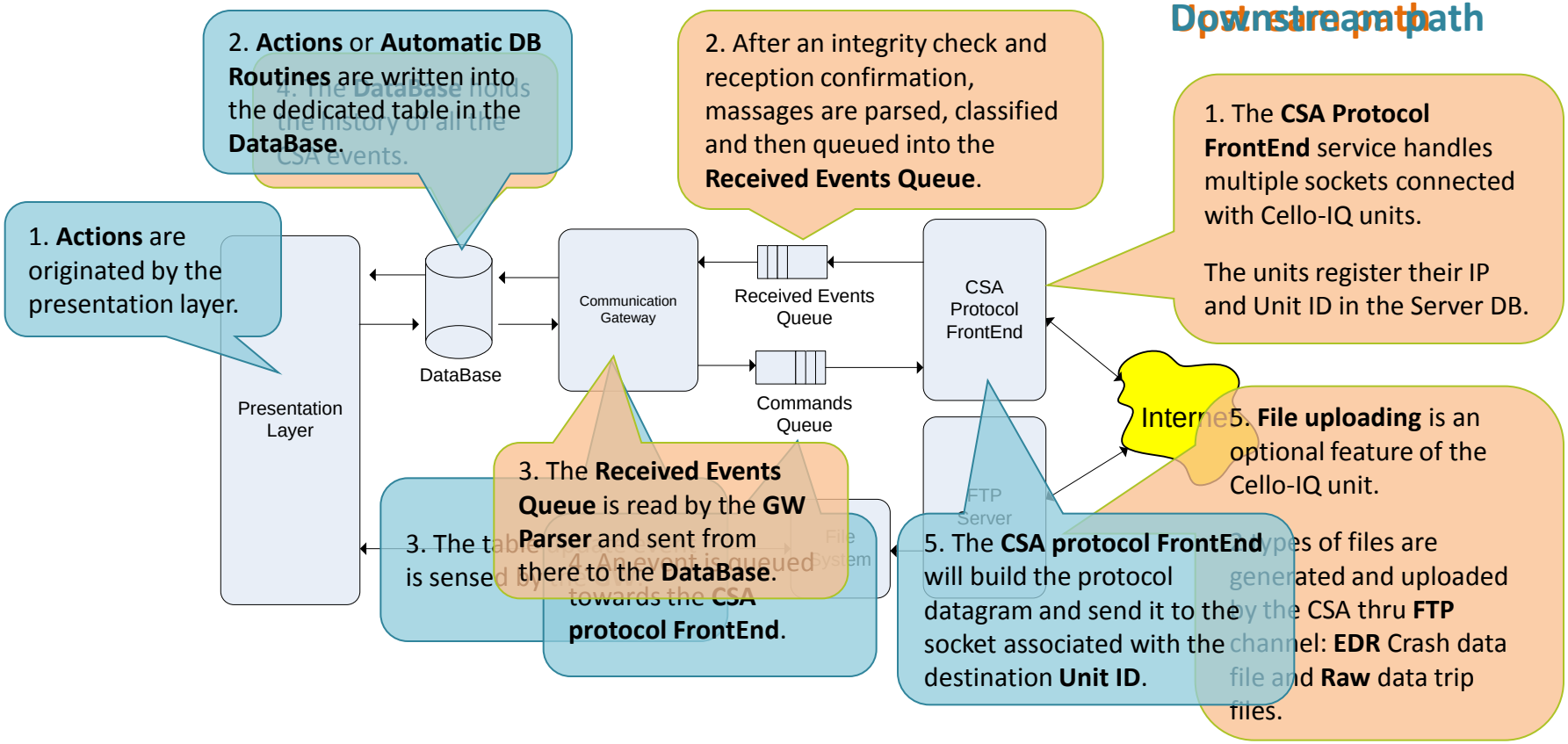
- **How to start Cellocator Integration?**

- Cellocator support self-implemented integration by providing:
  - **Documentation** on the product and implementation procedures, including overview, programming manuals, protocols, code samples etc.
  - The **Evaluation Kit** provides an important tool, helping the integrator to learn the product's expected behavior and functioning, working with the unit simulators, recorded data samples etc.
  - Other Utilities such as unit simulators, recorded data samples, etc.
  - Customer support
  - Professional services

Server Side CSA Integration

- **Generic CSA server block diagram**

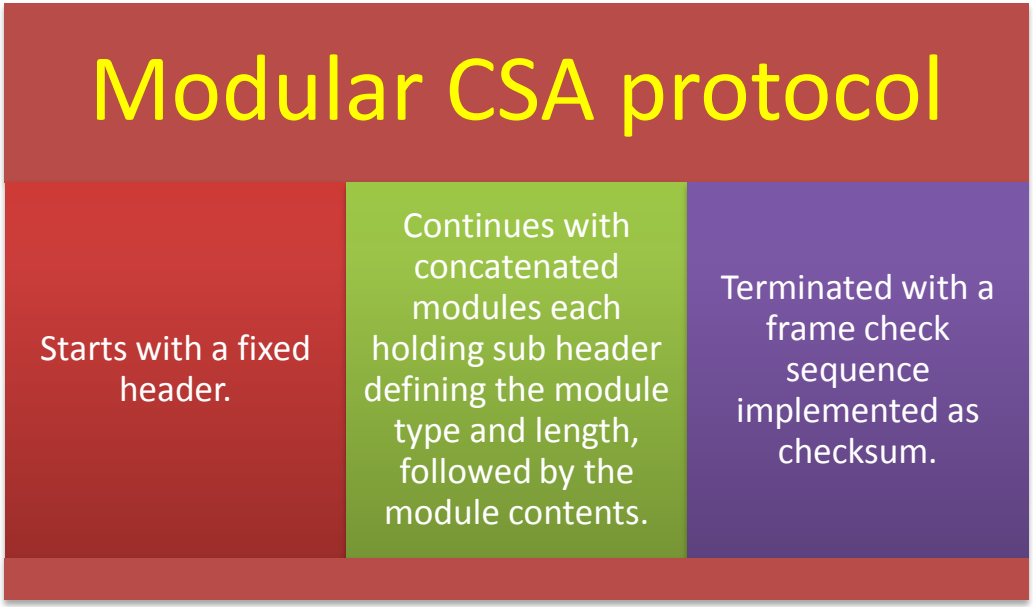
  - The block diagram describes a generic CSA server application intended to handle multiple Cello-IQ units

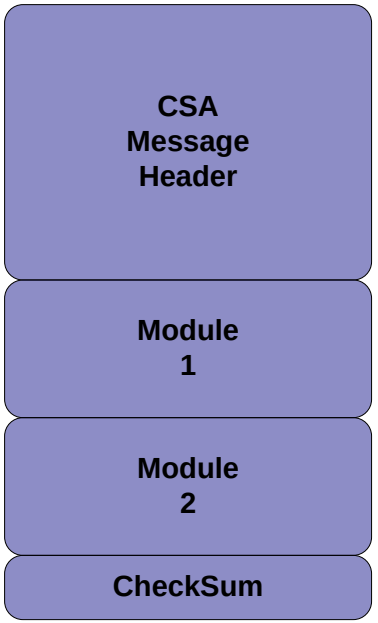


CSA Protocols

- **CSA uplink protocol structure and data flow**

❖ A modular protocol that supports variable length message





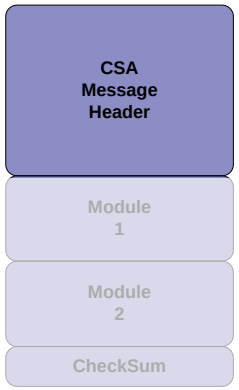
CSA Uplink Protocol Structure

- **CSA header format**

1	System code, byte 1 – ASCII “C”
2	System code, byte 2 – ASCII “S”
3	System code, byte 3 – ASCII “A”
4	Length of message from byte 6 to CS
5	
6	Message ID (Sequential numerator used by ACK mechanism. In replies contains the numerator of the command.)
7	
8	Message Type / Initiator / Direction (0 for CSA event)
9	Cello Unit’s ID (total 32 bits)
10	
11	
12	

Initiator	Direction	Protocol Version			Message Type		
0 – reply or ACK 1 - active	0 – inbound 1 - outbound	1					
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

1. Parsing CSA messages starts by verifying the message prefix. The prefix for all CSA messages is the 3 ASCII characters “**CSA**”.

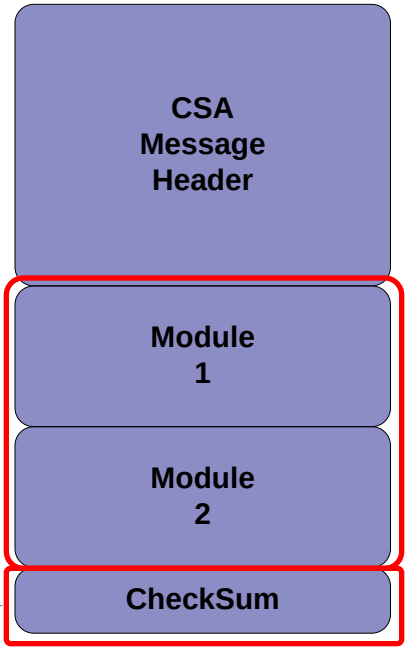


CSA Uplink Protocol Structure

- **CSA Module structure**

1	System code, byte 1 – ASCII "C"
2	System code, byte 2 – ASCII "S"
3	System code, byte 3 – ASCII "A"
4	Length of message from byte 6 to CS
5	
6	Message ID (Sequential numerator used by ACK mechanism. In replies contains the numerator of the command.)
7	
8	Message Type / Initiator / Direction (0 for CSA event)
9	Cello Unit's ID (total 32 bits)
10	
11	
12	

Initiator	Direction	Protocol Version			Message Type		
0 – reply or ACK 1 - active	0 – inbound 1 - outbound	1					
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0



2. The parser mechanism calculates the message checksum by summing the bytes ***following*** the length (bytes 4 and 5).

3. The 8 bits sum is then compared with the last message byte holding the received checksum.

4. After the message integrity is validated the server starts the module level parsing.

CSA Uplink Protocol Structure

- **CSA Module structure**

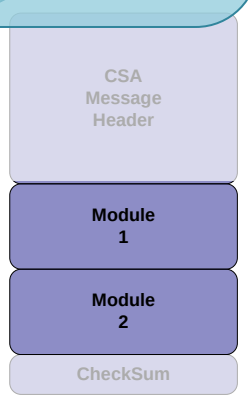
	Module Name 30 (**CSA Full Event**)
	Length of module - 46
	CSA Event Reason
	CSA Event Sub Reason
	CSA Event Numerator (or zero in case of reply)
	.
	.
	.

The detailed modules structure is part of the **Cellocator OTA protocol**. The **GW** parser should refer to the Protocol version bits in order to determine the expected modules are and the actual structure of these modules, which may change with the evolution of the CSA protocol versions.

5. The first byte describes the modules type, for example **"CSA FULL Event"**.

7. **CSA Full Event** has **Event Reason** and **Sub Reason** bytes describing the events details.

For example, **Hard Breaking** Event Reason is **"7"** and it's Sub Reason **"2"** shows it's severity, which is **Yellow Severity**.



CSA Uplink Protocol Modules

- **Common modules used to report maneuvers, scoring and statistics**

CSA full event (Module 30)

- The main report entity used by the CSA to report all sorts of CSA activities and triggering such as start / end of a trip, Ignition on or Off, IPUP, Go / Halt events and of course, any driving behavior event or maneuver detected and processed by the CSA.

ABC maneuver statistics (Module 31)

- If enabled, this module is concatenated to Module 30 of short term maneuvers (Brakes, Turns, Accelerations etc) and holds many maneuver attributes such as start and stop, location of the maneuver, average and max values of speed and accelerations, duration of the maneuver etc.

Trip statistics (Module 32)

- This module, if enabled, is sent upon end of Trip (Ignition off / Driver change) and holds many trip related valuable information like time driven, time idling, score of the trip, number of maneuvers of each type and severity etc.

Crash attributes (Module 35)

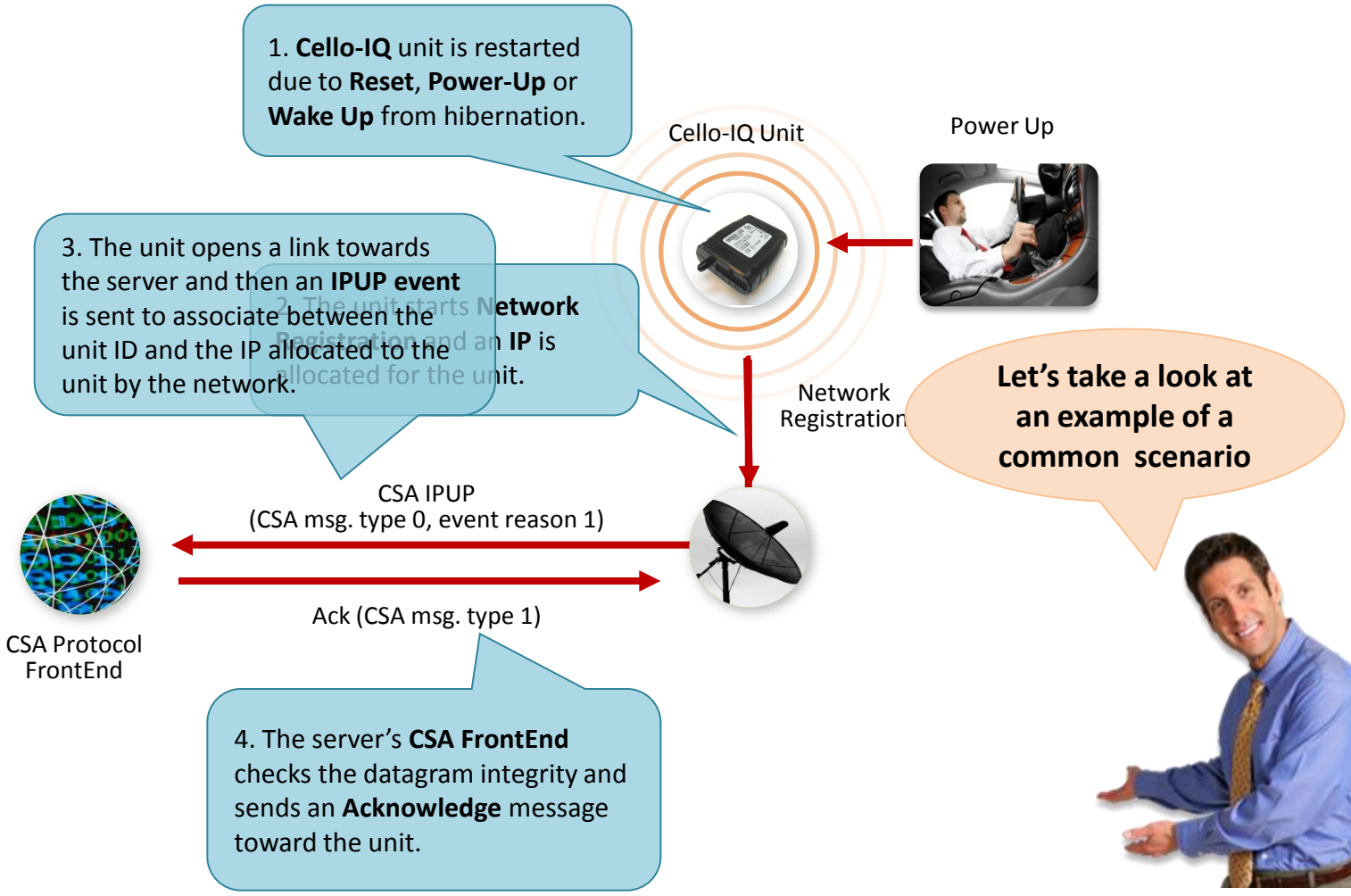
- Upon crash detection, either light or heavy crash, this module is generated and delivered through GPRS or SMS on top of an optional voice call triggering. This module holds Minimum Set of Data which is crucial in order to determine on the control center the severity and type of the crash, the attributes and identity of the vehicle, the location of the crash etc.

Continuous events statistics (Module 58-61)

- Reports various attributes of long term maneuvers such as over speeding, wrong gear (RPM), idling and off-road sessions. Valuable information such as the start / end location of the event, its duration, its extreme values and averages, and of-course the score are reported.

Typical CSA Communication

- **Typical CSA activity and reporting scenarios – for example IPUP**



Optional CSA Activity

- **Files upload by the CSA thru FTP channel**

Two types of files are generated and uploaded thru FTP channel: **Crash Data** file for EDR (emergency data reconstruction), and **Raw Data** trip files.


FTP Server


CSA Protocol FrontEnd

3. A **CSA event** is sent.

Cello-IQ Unit


Crash Event

Accident

GPS Fix

Crash Occurred
(CSA msg. type 0, event reason 2, Crash severity)

Ack (CSA msg. type 1)

2. The unit continuously logs the **Acceleration** and **GPS stamps** and when an accident is detected, a higher resolution sampling starts till the buffer is filled.

4. The accident "Story" captured is sent towards the **FTP** server.

Crash data ready to Upload
(CSA msg. type 0, event reason 3)

Ack (CSA msg. type 1)

Raw Data trip files are sent at the end of the trip when ignition off is detected. The files consist of **Maneuver Accelerations** and **GPS stamps** according to the severity requested by the user in the **CSA PL**. The accumulated files are being sent to the **FTP server** till all the files are uploaded or a time limit has been reached.

Crash file upload

File Upload Completed
(CSA msg. type 0, event reason 17, File upload status)

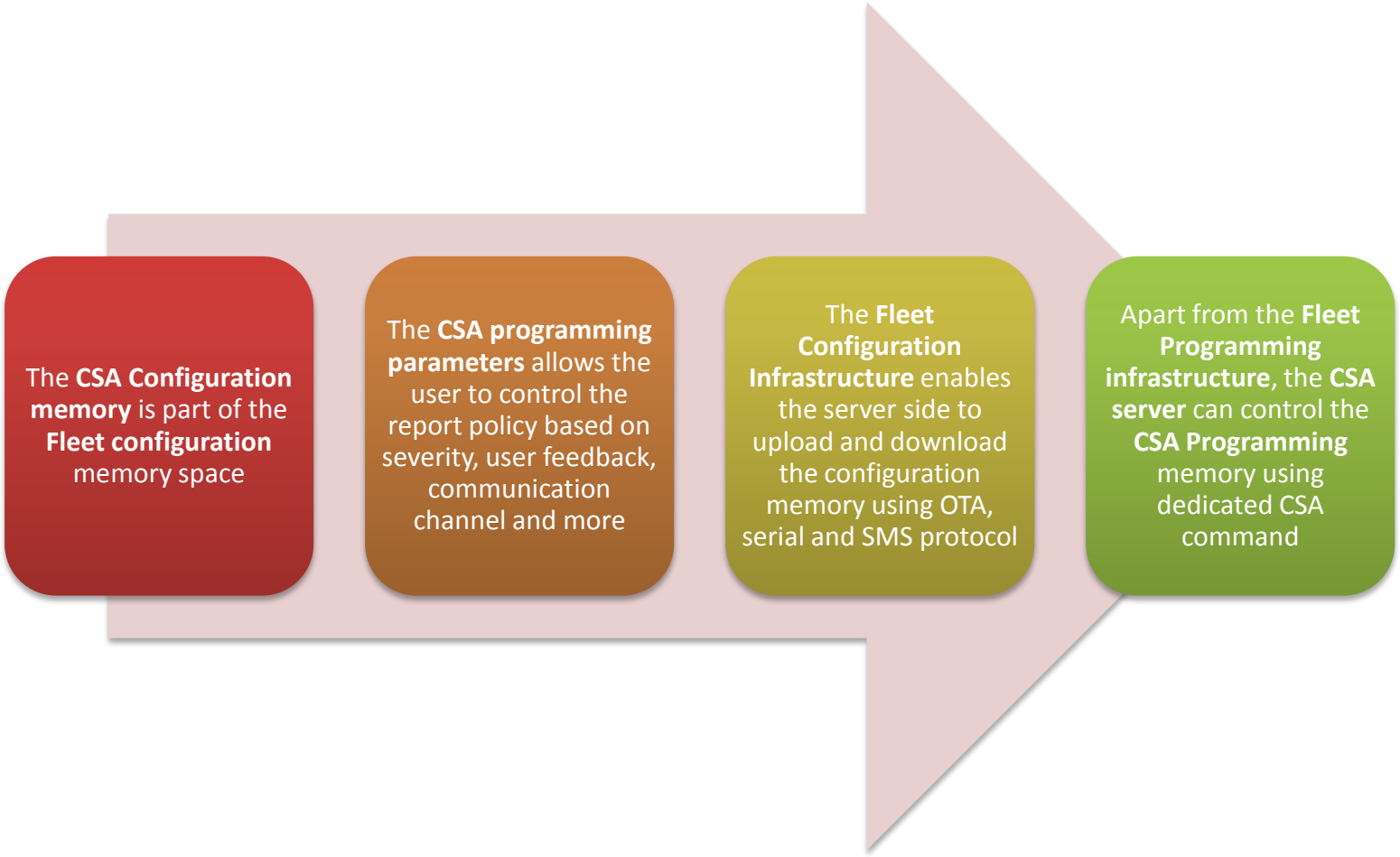
Ack (CSA msg. type 1)

POINTER

Cellocator Division
Pointer Telocation Ltd.

Downlink Protocol Management Principles

- **CSA programming memory**



Downlink protocol

CSA commands

- Programming module structure

	Module's ID (**10 - Programming Frame**)
	Length of module
	Programming command numerator
	• Action byte (Read/Write/Lock/Unlock) • 0 for Read command • 1 for Write command • 2 for Lock command (an infrastructure - currently not used) • 3 for Unlock command (an infrastructure - currently not used)
	The first address
	Length of data
	The data (in case of Read programming - single byte of Zero)

- Programming commands sent via **CSA protocol** will be saved into the **CSA space** of the nonvolatile configuration memory
- The configuration will be effective as soon as the unit will be restarted via **Fleet Reset OTA command**

Downlink protocol

- **CSA commands**

  - Reply programming module structure

	Module's ID (11-Reply Programming Frame)
	Length of module
	Reply to Programming command numerator
	Status
	Status (Success – 0 Failure – 1) / Failure ID
	Bit 7 / Bit 6 / Bit 5 / Bit 4 / Bit 3 / Bit 2 / Bit 1 / Bit 0
	Spare
	The first address
	Length of data (Zero - In all cases except Reply to Read Programming)
	The data (only in case of Reply to Read)

  - The same **CSA protocol interface** can request loading the CSA configuration for server side storage

Application Layer Integration

- **Two issues in the application layer integration require additional attention:**
  - Server side Driver Ranking
  - Server side Over-Speeding monitoring

Application Layer Integration

- **Server side Driver Ranking**

- Cello-IQ delivers on-board processing via maneuver and trip scoring logics
- However, for driver scoring process, the integrator should
  - Take into account relative distance (recommended)/time driven by a specific driver
  - Compare to another driver/certain population of drivers (Group, the whole fleet, the whole monitored population, etc) within the same vehicle category
- This weighting process is essential in order to reflect the frequency of a driver's wrong behavior over traveled distance or over time

Please consult **Cello-IQ Integration Manual, chapter 5** For recommended guidelines for the calculation of a driver's relative score.

Application Layer Integration

- **Server side Over-Speeding monitoring**

- For Fleet Safety Operations in which the TSP wishes to implement backend **Over-Speeding Events Management**

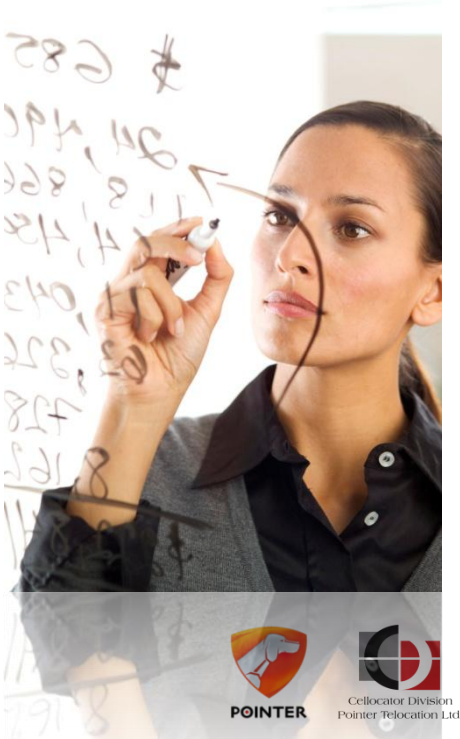
- **Over-speeding Events Management** is based on **Speeding Profile** provided by Cello-IQ with a cross-reference with **road-specific speed limit** values obtained from a GIS database

- Alter **Speeding-Free Trip Scores** received from the Cello-IQ (Module 32 in CSA protocol), in order to combine it with the **Over-Speeding Event Scores**, generated by the server side

- The **Integration manual** propose a logic to be executed on the backend upon the "end of trip", whenever the speeding profile is enabled in the PL, and thus speeding events are not taken into calculation in the trip safety score

Additional Information

- You can also find more information on the DBM Self Implementation at:
  - Cellocator Integration Manual: http://www.cellocator.com/knowledgebase/cellocator-cellofamily/cello-iq/
  - Cellocator Wireless Protocol manual:



Cellocator Cello-IQ – Let’s take a ride

POINTER

Cellocator Division
Pointer Telocation Ltd.