Deliverable D3.3
Data management in euroFOT

Version number	<Version 1.0>
Dissemination level	<PU>
Lead contractor	<CHAL>
Due date	<31.08.2011>
Date of preparation	01.08.2011

Authors

Selpi, Chalmers tekniska högskola

Samuel Borgen, Chalmers tekniska högskola

Jonas Bärgman, Chalmers tekniska högskola

Erik Svanberg, Chalmers tekniska högskola

Marco Dozza, Chalmers tekniska högskola

Rasmus Nisslert, AB Volvo

Catarina Norell, AB Volvo

Jordanka Kovaceva, Volvo Cars Corporation

David Sanchez, CTAG

Marga Saez, CTAG

Clement Val, CEESAR

Jörg Küfen, IKA

Mohamed Benmimoun, IKA

Barbara Metz, IZVW

Project Co-ordinator

Aria Etemad

Telematics & Navigation Research

Ford Research & Advanced Engineering Europe

Phone: +49 241 9421 246

Fax: +49 241 9421 301

Email: aetemad1@ford.com

Ford Forschungszentrum Aachen GmbH

Suesterfeldstr. 200

D-52072 Aachen

Germany

Revision and history chart

Version	Date	Reason
0.23	2011-08-10	Version for internal review
1.0	2011-08-31	Final version for submission

Table of contents

List of Tables

List of Figures

Executive summary

Data management is essential for a successful field operational test or naturalistic driving study. The complexity of data management tasks and thus the resources needed are often underestimated. This deliverable, "D3.3: Data management in euroFOT", is a public deliverable of subproject 3 (SP3: "Data Management") of euroFOT. Its purposes are: 1) to describe the data management solutions (the data acquisition system, database and storage, and the base of analysis tool) used in euroFOT; 2) to summarise how we came to choose those solutions; 3) to summarise the quality assurance checks performed throughout the whole data management chain; 4) to share our experiences and the lessons learned.

The main contributions from D3.3 to euroFOT and a wider community include:

- Five state-of-the-art data acquisition systems that have been successfully used for FOT: 1) CTAG Datalogger II (CAN-only logger), 2) CEESAR videologging system (a combined CTAG Datalogger II and video logger based on Nexcom VTC3300, with radar, eye tracker and lane tracker), 3) SAFER-euroFOT DAS (an integrated CAN and video logger based on Nexcom VTC6100, with external accelerometer and eye tracker), 4) BMW DAS (a CAN and video logger based on network attached storage, with two radar sensors), and 5) Daimler DAS (a CAN, video and audio logger);

- Three implementations of web-based DAS-vehicle monitoring tools for quality assurance;

- The required infrastructure and services for data storage, including pre-processing and uploading tools. The solutions chosen vary depending to the expected total size of data. The video data are stored as files in file management systems. The non-video data are generally stored in relational databases; Oracle, MySQL, and MS Access are used. At some partners, the non-video data are also stored as MATLAB files in the file management systems, to allow analysis through database and/or direct with MATLAB.

- Three implementations of the base data analysis tool that allows researchers to visualise time-based data from multiple sources and manually or automatically annotate the data;

- A set of requirements for the design of data acquisition systems and sensors. Fulfilling this new set of requirements, in a production scale, would further advance the state-of-the-art in data acquisition systems and sensor technologies and their applicability for FOTs;

- 47 lessons learned.

    Working methods that proved positive include: using common templates for gathering requirements; having specialised/focused groups to enable joint efforts in performing specific tasks; separating the handling of incoming disks/data from pre-processing stage; using modular approach for developing the base data analysis tool.

    Common DAS-hardware issues include: broken compact flash and SD cards, failing connectors and loose cameras.

    Recommendation made include: use industrial grade compact flash cards and SD cards; give extra attention when installing connectors and cameras; and finally be open to the potential benefits of using database features like partitioning and compression to enhance the standard relational database, using proper data warehousing, or even to other technologies (e.g. column-oriented databases, Hadoop database (HBase) or data warehouse for Hadoop (Hive)) for the management and analysis of FOT data of the scale of this project or larger projects.

1   Introduction

The goal of subproject 3 (SP3: ``Data Management´´) of euroFOT is to provide robust and flexible data management solutions for euroFOT. Specifically, the objectives of SP3 (euroFOT, 2008) are to:

1. Provide the best suited data acquisition systems (DAS) for euroFOT;

2. Provide the required infrastructure and services for data storage;

3. Develop analysis tools and general quality assurance;

4. Support SP2 and SP5 in installation and maintenance strategies (see also Figure 4).

These objectives of SP3 are mapped into six work packages within SP3 (Figure 1). The interfaces between the three main work packages are shown in Figure 2.
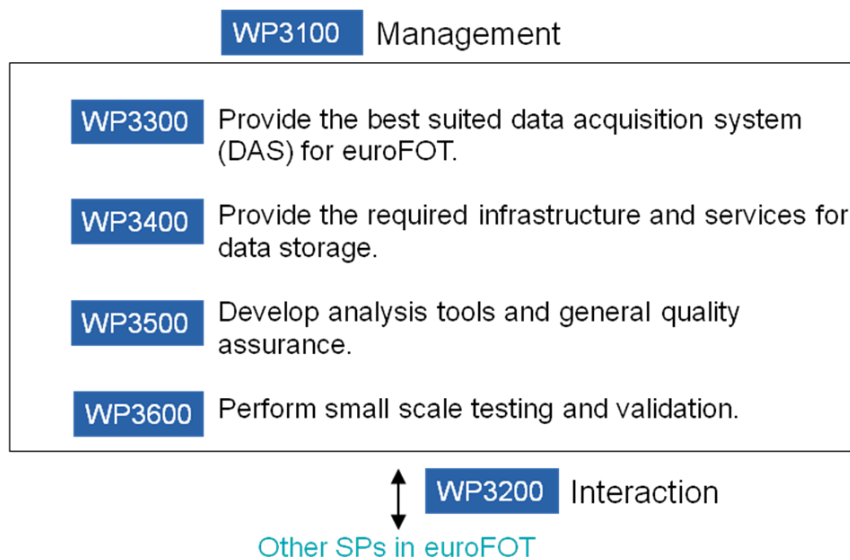
Figure 1: SP3: objectives and work packages.
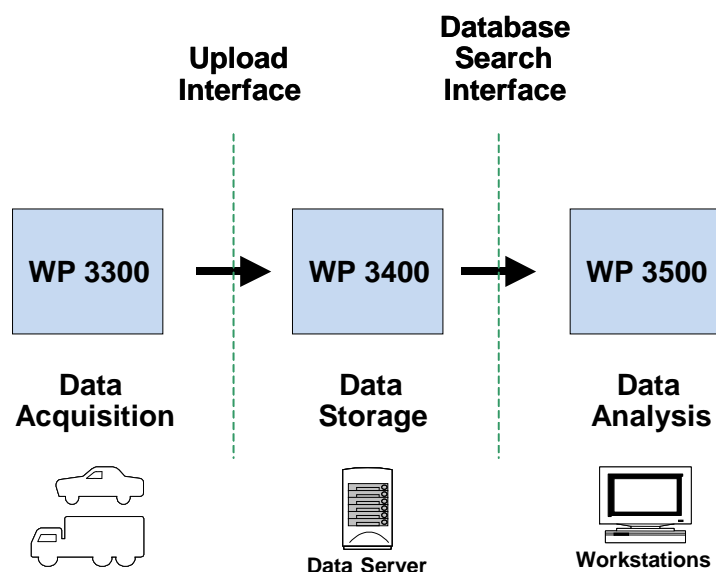
Figure 2: Data management chain considered in euroFOT (SP3).

The role of SP3 within the FOT chain (introduced in FESTA) is shown in Figure 3. In order to achieve the objectives of SP3, collaboration with other subprojects within euroFOT was essential (see Figure 4). Within euroFOT, functions and hypotheses are the main focus of SP2, while performance indicators, study design, and measures are the focus of SP4. Thus, inputs from SP2 and SP4 were essential to make sure that the chain from hypotheses to requirements specification for DAS is preserved, that is to say that at any point in time the reasoning of why certain signals or certain external sensors were necessary is clear. Furthermore, SP3 also collaborated with SP5 and SP6, as these two subprojects are direct users of SP3. SP5 is the subproject that is responsible for managing the field operational study (FOT) within euroFOT (including handling ethical and legal issues), and SP6 is responsible for conducting the analysis. Therefore, SP5's and SP6's needs were always valued and considered by SP3 to make sure that SP3 provides the right solution for the project.

This deliverable, D3.3, summarises the work that has been undertaken, mainly by SP3 partners, towards achieving the objectives of SP3. Specifically, the objectives of this deliverable are to:

- summarise the process we followed to select and finally made available data management solutions for euroFOT;

- describe the DAS solutions used in euroFOT;

- describe how we process and store the FOT data (including what kind of quality checks done);

- describe the tools used in euroFOT to extract, view, and annotate the FOT data;

- share the experience and the lessons we learned through this project from an FOT data management perspective.
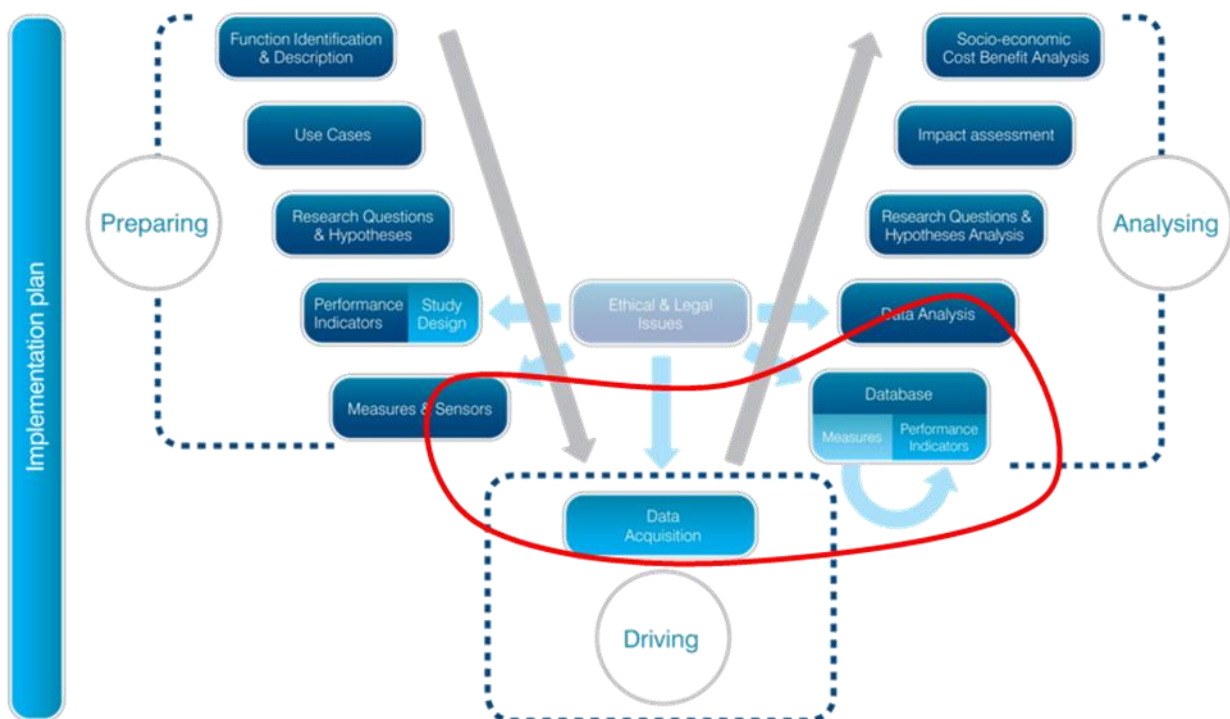
Figure 3: FOT-chain introduced in FESTA. SP3 role is marked with red curve.
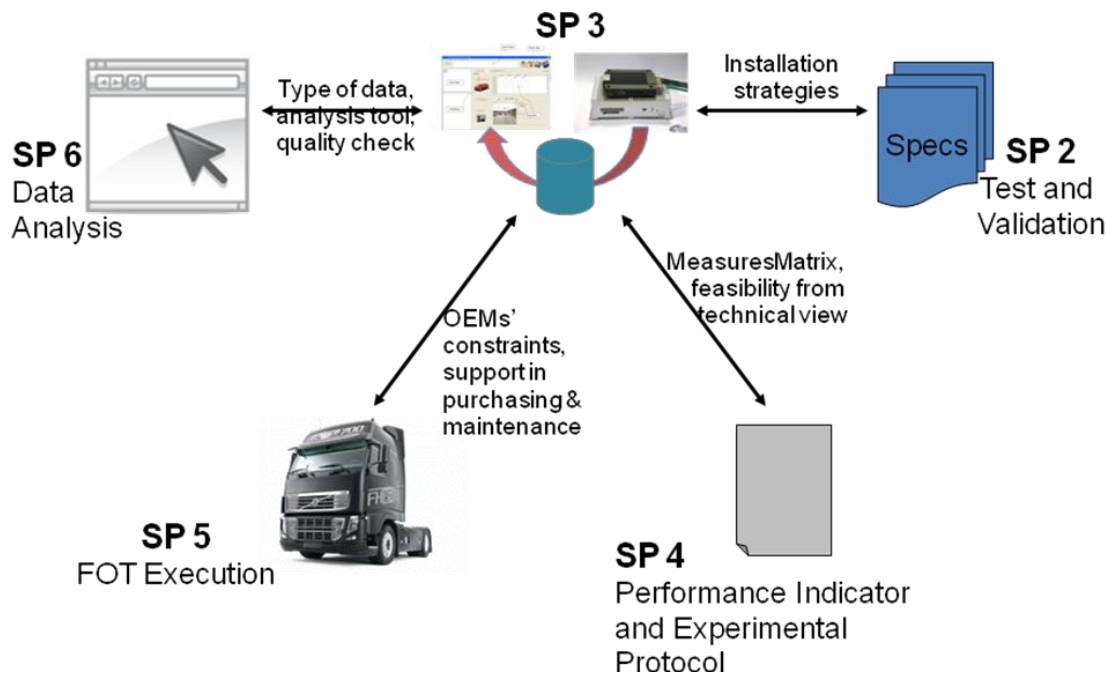
Figure 4: Collaboration between SP3 and other subprojects (SPs) in euroFOT.

1.1 Vehicle Management Centre (VMC)

The field operational test (FOT) in euroFOT is managed in four vehicle management centres (VMCs). A VMC is an entity which can be seen as the overall organisation responsible for management and logistics of FOT. The four VMCs and the partners involved in the VMCs are (see also Figure 5):

- French VMC: CEESAR and IFSTTAR with support from Renault. CEESAR is the partner that manages the French VMC database.

- German VMC is divided into two operation centres (OCs):
    - Operation Centre 1: VW/AUDI, FORD, MAN, and IKA. The data management in this OC is taken care by IKA.
    - Operation Centre 2: Daimler, BMW, and IZVW. The partner that manages the data management in this OC is IZVW.

- Italian VMC: CRF and POLI. The data management for this VMC is conducted by POLI.

- Swedish VMC: VOLVO, VCC and Chalmers. The Swedish VMC database is hosted at and managed by Chalmers.

A detailed description of each of the vehicle management centres is available from Deliverable D5.1 (Flament, Cavallo, & Hagleitner, 2009).

The participating partners in SP3 are Chalmers, IKA, Volvo, VCC, CEESAR, CTAG, ICCS, and Bosch. The German-2 OC and Italian VMC are not represented in SP3 (i.e., Daimler, BMW, IZVW, CRF and POLI are not member of SP3). Therefore, only a summary of the data management solutions used by German-2 OC and Italian VMC is presented here; more detailed descriptions of their work in terms of data management are available in other euroFOT deliverables.
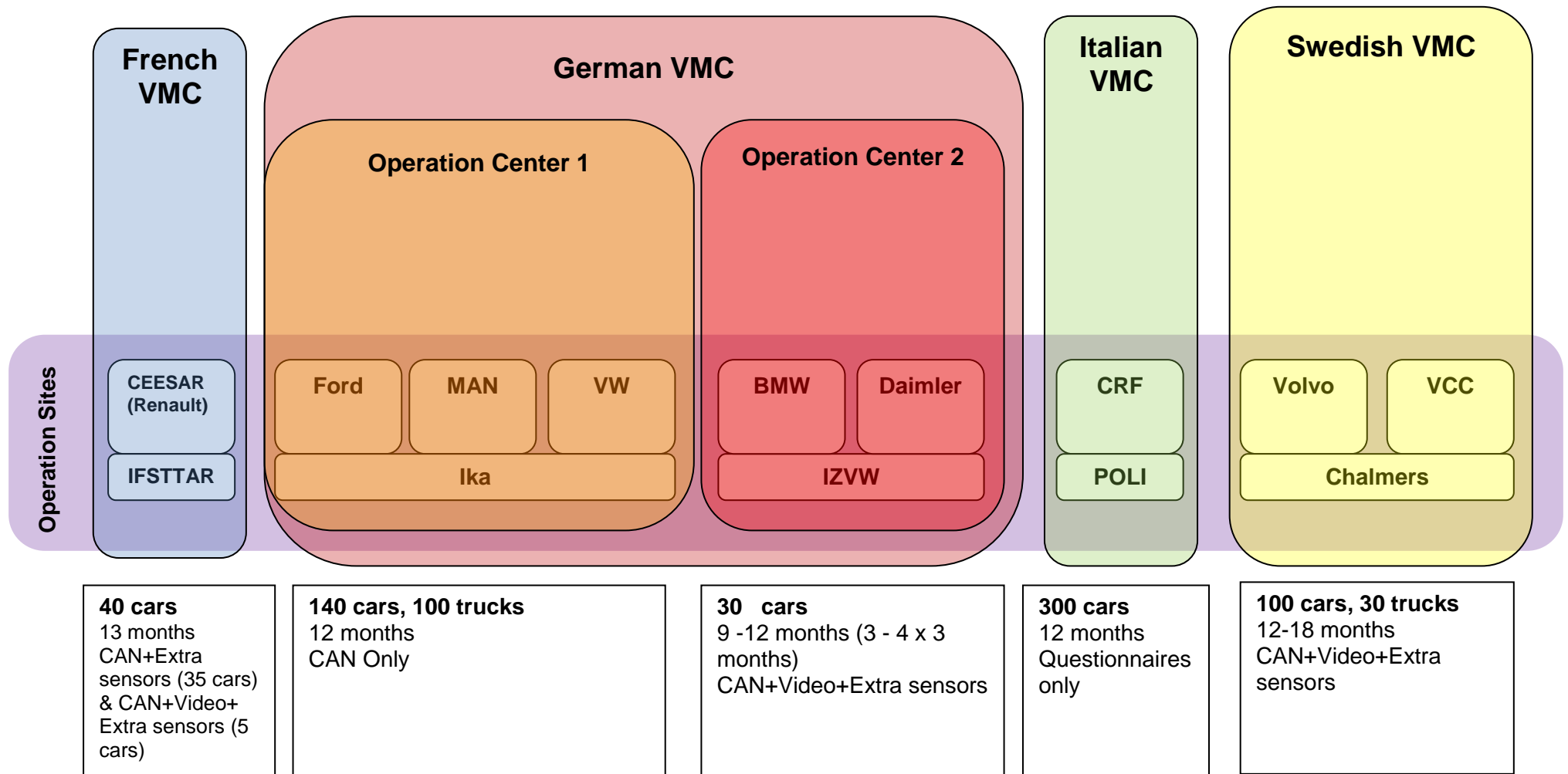
Figure 5: The VMCs in euroFOT and the FOT plan in July 2009.
This figure is adapted from D5.1.

1.2 Overall Way of Work

Each of the VMCs had planned to evaluate different set of safety functions (and thus different set of hypotheses). To answer the different set of hypotheses, each of the VMCs needed different types of data and thus different solutions for collecting, storing and processing the data. Furthermore, each of the VMCs had different constraints with regard to the kind of equipments that can be installed in the vehicles. All of these differences created a complexity. On the other hand, these differences give us a richer experience, for example in terms of using different systems, and richer lessons learned.

Common method(s) was used as much as possible. SP3 partners worked both at euroFOT level as well as at VMC level. The tasks that could be carried out together and/or could benefit more than one VMC were shared among partners, and done together at euroFOT level. However, due to the differences described previously, there were also some work i.e., to find specific solution(s) for each VMC, that have had to be done at the VMC level. This allows each VMC to concentrate their resources on different tasks. Internally, SP3 partners worked together on different parts of the data management chain. The interconnection and relation between the work packages inside SP3 and with other subprojects were coordinated at euroFOT level by SP3 leader and WP3x00 leaders. As a whole, SP3 provides common ground as well as specific solution(s) for each VMC.

1.3 Organisation of D3.3

This deliverable is organised into six chapters:

- Chapter 1 introduces the purpose of this deliverable, an overview of SP3 work, the chosen way of work, as well as the term vehicle management centre (VMC) which will be used throughout this deliverable;

- Chapter 2 describes the data acquisition systems used in the different VMCs;

- Chapter 3 describes how we process and store the FOT data;

- Chapter 4 describes the tools used in the different VMCs to extract, view, and annotate the collected data;

- Chapter 5 summarises our experiences and lessons learned in terms of data management in the euroFOT project and makes recommendations for future FOTs;

- Chapter 6 sums up the contributions to the state-of-the-art made by SP3 partners. This chapter also concludes this deliverable.

2   Data Acquisition System (DAS)

This chapter

- summarises the process that was used by euroFOT partners to select and finally made available the data acquisition systems in the different VMCs (Section 2.1);

- describes the data acquisition systems used in euroFOT (Sections 2.2);

- shares the lessons learned (Section 2.2).

Here, DAS is defined as the in-vehicle hardware and software that constitutes a complete logging system, including external sensors used for the purpose of logging necessary signals for FOT. In the deployment of field operational tests, data-logging is a key issue, and euroFOT is no exception. Success of the project relies, somehow, on the DAS performance; it should log all the information planned to be logged with no data-loss or corruption, be proven to work in different conditions, and be reliable for use in a long period (at least one year) by a large number of vehicles.

2.1 Method/Procedure Used

The steps that were followed throughout WP3300 work to provide best suited data acquisition systems for euroFOT can be summarised as follows:

1. Gather initial requirements and/or constraints from the different VMCs (or specifically OEMs) and identify necessary signals and sensors;

2. Gather technical requirements specification, considering safety, ethical and legal issues, as well as cost;

3. Identify possible DAS solutions, conduct market survey and evaluation;

4. Decide which DAS solution(s) to use and make the DAS ready for euroFOT.

Step 1 is summarised in Section 2.1.1. The highlights from steps 2-4 are summarised in Section 2.2.

2.1.1   From functions and hypotheses to signals and sensors

The initial requirements and/or constraints that had to be taken into account in the process of selecting suitable DAS are:

1) functions & hypotheses to be tested, performance indicators (PIs), study design, measures, sensors allowed/needed, ethical and legal issues as shown in Figure 3 (FESTA-consortium, 2008); The safety functions tested by the different VMCs are shown in Table 1.

2) number and type of FOT vehicles (car, truck) considered to be equipped, length of data collection considered, and restriction on type of data considered to be collected (if any).

A requirement or constraint to consider from the beginning is the type of DAS to be used by each VMC: 1) CAN only; 2) CAN + video; 3) CAN + video + extra sensor(s). However, the CAN + video is finally changed to CAN + extra sensor(s).

During the first year of the project, types of vehicle to be equipped (cars, trucks), how many for each type and the length of data collection period were revised to reflect the situation at that time (the FOT plan in July 2009 is shown in Table 2 and the lower part of Figure 5).

Table 1: Safety functions tested in euroFOT.

Function	German-1			Swedish VMC		Italian	French	German-2	
	FORD	MAN	VW	VOLVO	VCC	CRF	RENAULT	BMW	DAG
Forward Collision Warning and Brake Support	✓	✓		✓	✓				
Adaptive Cruise Control	✓	✓	✓	✓	✓				
Speed Limiter and Cruise Control							✓		
Lane Departure Warning		✓	✓	✓	✓	✓			
Blind Spot Information System					✓				
Safe Human Machine Interaction								✓	✓
Curve Speed Warning	✓								
Fuel Efficiency Advisor				✓					
Impairment Warning					✓				

Table 2: The FOT plan in July 2009.

Type of DAS	German-1			Swedish VMC		Italian	French	German-2	
	FORD Cars	MAN Trucks	VW Cars	VOLVO Trucks	VCC Cars	CRF Cars	RENAULT Cars	BMW Cars	DAG Cars
CAN only	100	100	40						
CAN + extra sensor(s)							35		
CAN + video + extra sensor(s)				30	100		5	15	15

Based on Table 1, a set of hypotheses for each function was formulated by SP2. From the hypotheses, SP2 suggested which performance indicators, controlled factors and variable factors to be collected during FOT study. Detailed functions specification and hypotheses that are evaluated in euroFOT are described in Deliverable D2.1 (Dozza, Kessler, Petersson, Rakic, Tattegrain-Veste, & Heinig, 2010).

The set of hypotheses as well as the suggested performance indicators were used by SP4 to determine the necessary performance indicators, situational variables, events, and experimental design. These performance indicators, situational variables, and events were then used by SP4 to derive a list of measures needed to answer the hypotheses. The list of measures was then used by SP4, in collaboration with SP3, to derive a list of signals suggested for collection. For simple cases, the signals are the same as the measures. While for other cases, a measure can be a function of one or more signals. The list of signals was then used by SP3 to find out availability of the signals from the systems on-board the vehicles. When necessary, external sensors were considered to get those signals that, for some reasons, are not available from the vehicle systems. The complete lists of performance

indicators and measures are described in Deliverable D4.1 (Kircher, Heinig, & Brouwer, 2009), and the experimental design is described in Deliverable D4.2 (Jamson, et al., 2009).

The list of signals suggested for collection includes:

- longitudinal driving behaviour such as speed, mileage, longitudinal translational acceleration, lateral translational acceleration, break force, brake pedal;

- lateral driving behaviour such as steering wheel angle, lateral position of the vehicle, yaw rate, direction indicator, lane marking type, lane width, number of lanes;

- data from radar and other vehicle sensors such as distance to vehicle ahead, wiper signal, headlight activation, high beam activation, fog light activation, selected/active gear;

- driver based system such as activation and setting of certain safety functions, warning activation;

- other vehicle system such as airbag status, ESV activation, ESV intervention, ABS intervention;

- engine information such as oil temperature, engine speed, and fuel consumption;

- video data such as video of driver's face, forward view, driver's feet, and cabin view;

- eye movement data such as gaze direction and eye closure;

- GPS data such as longitudinal and latitude positions, speed, direction of travel, time and date;

- weather data such as presence of rain, outside temperature, lighting condition, times of sunset and sunrise;

- geographical/map-based data;

- derived measures such as jerk, time headway, time to collision, and time to line crossing.

Based on the list of signals, the following extra sensors were considered:

- eye and head tracker;

- GPS;

- accelerometer and yaw-rate sensor;

- radar/headway position sensing;

- lateral position sensing.

2.2 DAS Solutions Used in euroFOT

An overview of the data acquisition systems used in the different VMCs is shown in Table 3.

Detailed descriptions of the chosen DAS solutions used in each VMC are provided in Sections 2.2.1 to 2.2.5. German-1 operation centre is presented first then French VMC and then Swedish VMC; this sequence was chosen as there is a gradual increase in complexity of the data acquisition systems to be used in these VMCs. German-2 operation centre is presented after the previous three; this OC is the only OC in euroFOT that studied SafeHMI. The Italian VMC is presented last, as this VMC does not actually use sophisticated DAS as the other VMCs.

Table 3: The main features of the data acquisition systems used in the different VMCs.

| Category | German-1 OC | French VMC | | Swedish VMC | German-2 OC | |
		Light instrumentation	Heavy instrumentation		Daimler	BMW
Main component	CTAG Datalogger II	CTAG Datalogger II	CTAG Data logger II and Nexcom PC (VTC3300)	Nexcom PC (VTC6100)	PROVEtech:VA	ReadyNAS 2100
Number of CAN channels collected	4	2	4	4	4	4
Operating System	Embedded Linux	Embedded Linux	Embedded Linux, Windows XP	Windows XP Embedded	Linux	Embedded Linux
Storage device	SDHC (up to 8 GB)	SDHC(up to 8 GB)	SDHC (up to 8 GB) and 80GB automotive hard disks (for video)	compact flash card (16 GB) + 500GB office hard disks	3 USB-disks	4 x 1TB hard disks
Cameras	-	-	4 (dashboard view, forward, rear, feet)	For cars: 4 (forward view, rear view, driver, feet); For trucks: 4 (forward view, blind spot, driver/cabin, feet)	4 (driver face, forward view, rear view, navigation system)	4 (driver face, forward view, rear view, and centre stack to see driver hands and legs)
Radar	Existing vehicle radar (via CAN)	Long range ACC radar (200m)	Long range ACC radar (200m)	Existing vehicle radar (via CAN)	Existing vehicle radar (via CAN)	2 Short range radar sensors
GPS	Built-in-the data logger	Built-in-the data logger	Built-in-the data logger	External GPS	External GPS and existing vehicle GPS	Existing vehicle GPS

| Category | German-1 OC | French VMC | | Swedish VMC | German-2 OC | |
		Light instrumentation	Heavy instrumentation		Daimler	BMW
Lane tracker	Existing vehicle information (via CAN)	-	Mobileye	Existing vehicle information (via CAN)	Existing vehicle information (via CAN)	Existing vehicle information
Accelerometer	Existing vehicle information (via CAN)	Existing vehicle information (via CAN)	-	External two-axis accelerometer with additional yaw sensor		
Eye tracker	-	-	One camera system	One camera system	-	-
Audio functionality, route identification	-	-	-	-	Driver interaction box (for audio comments, and route identification) and microphone	-
External Navigation device	-	-	-	-	Added for certain period	Added for certain period
Data transfer	Wireless (CAN data)	Wireless (CAN data)	Local Network transfer	Wireless (status information), disk pick-up (CAN+video data)	Wireless (status information), disk pick-up (CAN+video data)	Wireless (status information), disk pick-up (CAN+video data)
Encryption	Yes	Yes	Yes	Yes	-	-
Compression	Yes	Yes	Yes	Yes	Yes	-

The task to decide which DAS solution(s) to use and make the DAS ready in each VMC was done at VMC level. However, as much as possible, the work to gather technical requirements specification, to identify possible DAS solutions, and to conduct market survey and evaluation were done together. Here, we give some examples of those joint tasks.

Collaborations between VMCs

Generally, several specialised/focused groups were formed to enable joint efforts in performing certain tasks, for example for investigating some sensors like eye trackers, radar, cameras, and map providers. The possibilities to use the same eye tracker, cameras and other extra sensors were also investigated. This was also done in the hope to get a better price per unit.

German-1 OC and French VMC conducted the market survey on data logger together, and finally chose the CTAG data logger. Since then, the evaluation of CTAG data logger was led by German-1. This was to allow French VMC to focus their efforts on finding and evaluating video loggers and sensors.

To fulfil euroFOT requirements specification (for both German-1 OC and French VMC), further development on CTAG data logger was needed. In this case, CTAG worked together with German-1 OC and French VMC in finding the best way to fulfil the requirements specification.

Regarding external sensors, eye and head tracker, radar/headway sensing, lateral positioning as well as cameras are of interest to both French and Swedish VMCs. Therefore, these two VMCs worked together in defining requirements specification for those sensors, conducting market survey, as well as in evaluating some of the possible solutions.

When compared to other VMCs, the Swedish VMC has more experience on video acquisition. Furthermore, the Swedish VMC has done extensive market survey on both dedicated video logger as well as on vehicle PC which can also be used as video logger. The French VMC has found that collaboration with the Swedish VMC in discussing requirements and candidate sub-components of the video logging system has proved useful.

Accelerometer and yaw-rate sensors were considered by both French and Swedish VMCs. The two VMCs worked together with Bosch to develop the requirements specification on these sensors and in identifying possible solutions.

Collaborations with other SPs

Regarding interaction with other SPs, SP3 has contributed to the task of feasibility ranking of SP2-hypotheses; input from SP3 was made from the technical point of view. This task was coordinated by Chalmers, but each VMC gave its input for this task. In addition to collaboration with SP2, SP3 also worked together with SP2 for in vehicle adaptation, for instance it is important for both SP3 and SP2 to make sure that the DAS can fit in the specified compartment in the vehicle where the DAS was planned to be put.

The measure list from SP4 was first assessed in each VMC within SP3, then in a meeting involving SP3, SP4 and SP6 to clarify any open issues like unavailability of signal(s) for certain measure(s) and to discuss alternative solution(s) that might work for all VMCs. In the latter meeting, representatives of each VMC within SP3 were present. Overall this collaboration was to make sure that SP3, as a whole, can provide all the necessary measures that SP4 identified as necessary to answer hypotheses defined in SP2 within available budget.

Safety, installation, insurance, and maintenance issues related to the installation of the DAS in the vehicles were some of the topics of interaction between SP3, SP5, and SP2. Concrete results from the interaction are: 1) the installation strategies and verification, which was used by SP2 for pilot installation and SP5 for FOT installation; 2) the maintenance strategies like the use of web-based monitoring tool, procedure to deal with DAS related problems during data collection, and procedure for data handling.

2.2.1   German-1 OC

2.2.1.1 Initial condition

Loggers / Vehicles	CAN Only	CAN + Extra sensor(s)	CAN + Video + Extra sensor(s)
Ford Cars	**100**	-	-
MAN Trucks	**100**	-	-
VW/AUDI Cars	**40**	-	-

The numbers above shows the plan in July 2009. Only 60 MAN trucks finally joined the FOT.

The main requirements/constraints affecting DAS choice in German-1 Operation Centre and French VMC (for the CAN-only logger):

- There must not be any integration or use of additional sensors (beside the DAS itself).
- From the vehicle itself, only data available on CAN-busses can be collected.
- GPS information (location of vehicles) is not available on CAN-bus but must also be collected.
- DAS should be a small device that can be integrated in the vehicles without significant noticeable modifications to the vehicle.
- Drivers should not be bothered with change of hard discs, configuration of DAS, or upload of data.
- DAS should be remotely configurable.
- DAS must be able to store collected data on a temporary storage on the DAS and when possible send the logged data (autonomously or upon request) to the server via any kind of WIFI or GPRS communication.
- DAS should have ability to compress and encrypt the collected data.
- DAS must be reliable and applicable for automotive application (high durability with regard to vibration, temperature, etc)

Based on these requirements the following hardware components need to be integrated in the DAS:

- CAN-Interfaces (at least for 2 CAN busses)
- CPU unit
- GPRS modem
- Integration of SIM card (upload via GPRS)
- Integration of hard disc
- GPS receiver
- GPRS antenna
- GPS antenna
- Power supply unit

With all of these requirements added to the common requirements, a market survey was carried out to find a suitable DAS solution for German-1 OC and French VMC (for the CAN-only logger). During the market survey, several data acquisition systems were investigated by gathering all available information about these systems. Thereby several aspects were examined. Apart from the requirements on the DAS, the costs were also considered. Finally all gathered information was documented onto a database. Based on this, the most suitable DAS (requirements, costs, etc.) was selected.

2.2.1.2 Chosen solution

From the requirements defined in Section 2.2.1.1, a market survey was performed and several DAS fulfilling the requirements were found.

Table 4: Result of market survey on CAN-loggers.
Items are colour coded; green (fulfil requirements), yellow (not enough information could be found), and red (did not fulfil requirements).

Provider	Name
CTAG	Datalogger II
BMCM	ePCII-LOG
CAESAR	QIC minilog
CANopen	USB-CANlog
CSM	Unican
Dreyer + Thimm	Mcombox
EXXOTEST	DLC-MUXDIAGII-C
IMC	imc busdaq2
IMC	imc busdaqX
IMC	imc busLOG
IMC	imc C1
IPETRONIK	M-LOG
IPETRONIK	S-LOG
IXXAT	CANcorder MMC
Kvaser	Kvaser memorator
Micro movement	MAHTechS M5X-PRO DAS
Nyes	Muxlog R
Nyes	MUXy fleet
racelogic	VBOX III
Seratec	M-TRAS
yes-GATE	FMS-500
yes-GATE	VDL-1000
SYSTEC	USB-CANmodul2
VECTOR	CAN CASE XL LOG
VECTOR	CANlog
VECTOR	Multilog

The best cost benefit price was found for the CTAG Datalogger II, which was selected as the DAS of the German-1 OC after first discussions with CTAG. This DAS integrates all needed hardware in a very compact design that can be easily integrated in the vehicles. The data logger is presented in Figure 6.
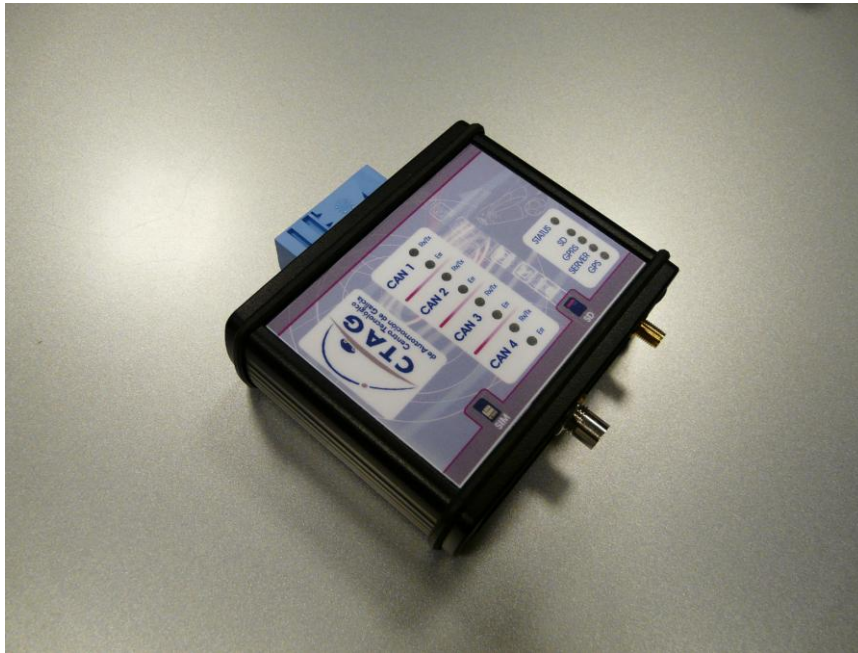
Figure 6: CTAG's Datalogger II

CTAG's DataLogger II is a storing device basically designed to receive and transmit CAN data by 4 channels and also to log the information onto an SD card. Apart from this main feature, the data logger can be located by a GPS receiver and is able to keep diagnostic communications with several ECU's at a time. The data logger can be controlled with a server application (Internet-based solution) for remote fleet management (tracking, logging, data logger configuration) on real time.

The main features of CTAG Datalogger II can be summarised as follows:

- 4 CAN 2.0B Channels Logging
- GPRS Quad Band Modem
- GPS receiver
- CAN baud rate configurable by user
- Built-in Real Time Clock (with internal battery)
- SD & SDHC compatible
- USB 2.0 High Speed configured as MSC
- Configuration application for logging parameterization and data conversion
- Supports DBC databases for signal selection
- Firmware upgrade through GPRS
- Led functional indicators

Some specific or unique features that this datalogger is providing to euroFOT are:

- Contact (KL15) behaviour emulation with very low consumption (< 0,1 µA when KL15 off)
- Python scripting enabled (can sending, traffic analysis, possibility to implement other services according to ISO 14229)
- Sample Server Application (Internet based solution) for remote fleet management (tracking, logging, datalogger configuration) provided.

- Data output format compatible with vector tools.

Basically a modem with GPS and GPRS capabilities was integrated within the Datalogger, but around it a practically whole brand new architecture was built up. The storage capacity was improved with the SD/SDHC compatible module. The modifications include a new automotive cable and modification of the server tool as well as configuration tool.

2.2.1.3 Final implementation to fulfil euroFOT requirements

Before the euroFOT project started, CTAG had already Datalogger v1.0. To fulfil the euroFOT requirements, a number of modifications and new features have had to be made. These modifications can be divided in HW and SW upgrades.
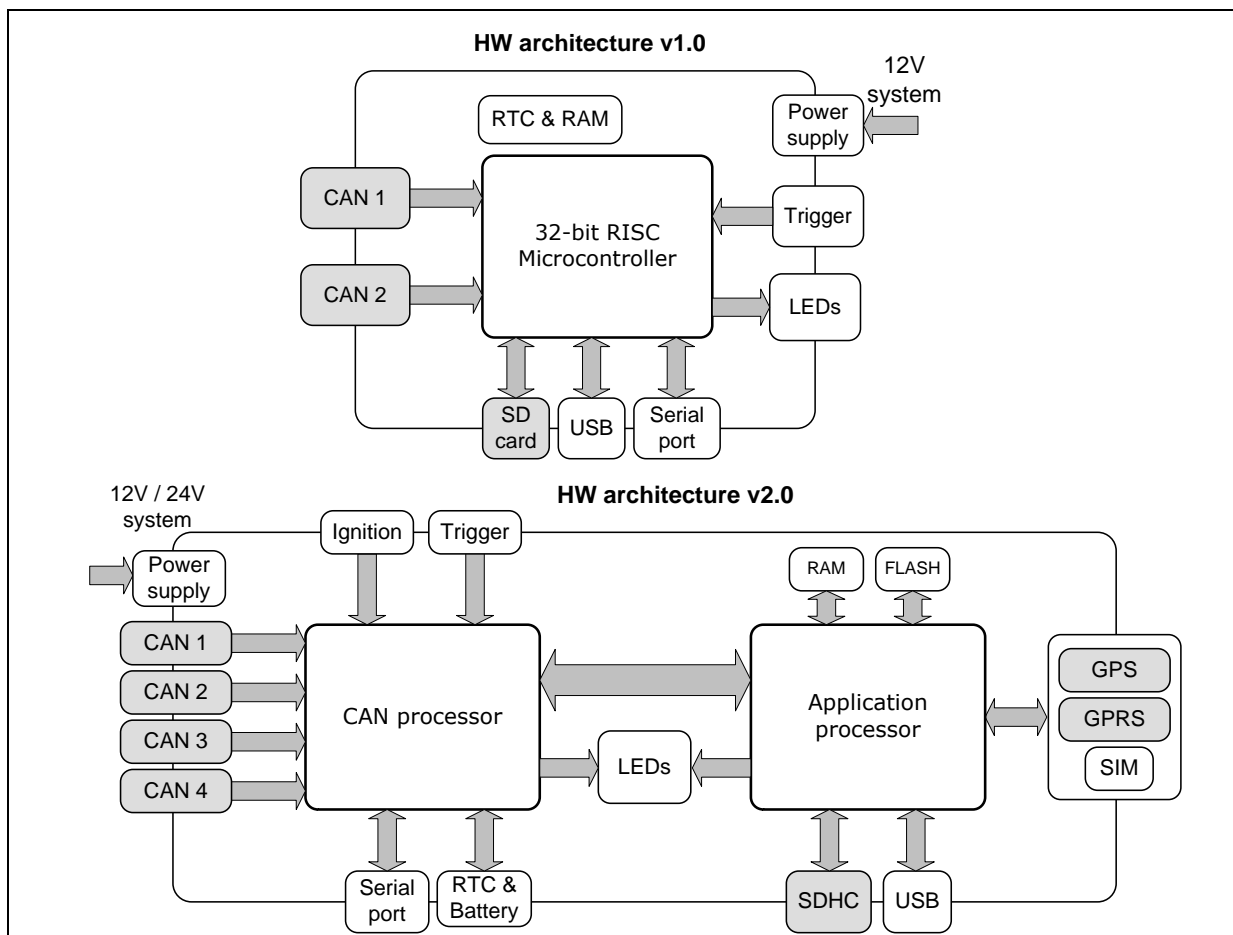
Figure 7: HW Architecture for CTAG Datalogger v1.0 and v2.0.

HW upgrade for Datalogger v2.0 (CTAG Datalogger II) to fulfil euroFOT requirements

The following modifications had to be implemented in the CTAG Datalogger in order to meet the new HW requirements (for comparison, see Figure 7):

- A new architecture was implemented, with 2 microcontrollers: CAN microcontroller (extending up to 4 CAN channels) and Application microcontroller.

- GPRS and GPS connexions (XT65 Siemens). The Datalogger v1.0 did not have any wireless external communication module, neither GPS receiver.

- SD and SDHC compatible. In version 1.0, there was a limit of 2 GB in the internal storage capability. With the new 2.0 version, SDHC cards can be used, boosting the off-line performance of the device.

- USB 2.0 High Speed (480 Mbps)

- New CAN and supply connectors (but still not automotive)

- Configurable LED indicators

- New automotive compliant connector for CAN and power-supply. A new automotive connector was integrated in order to fulfil the high level OEM requirements. In order to guarantee suitable reliability, a standard automotive connector was required. For this reason, the new version of the device is equipped with a micro quadlock TYCO connector shown in Figure 8. Micro Quadlock TYCO connector for:
  - o 4 CAN channels
  - o Power supply and ignition signal (KL15)
  - o Trigger

- Two power-in versions are available: 12V and 24V.

- Consumption reduction (KL15 mode): The high energy consumption was solved by using the ignition signal (KL15) for the power supply, i.e. the data logger will be supplied with energy only when the engine is on. In the current version of the Datalogger, consumption in sleeping mode or "idle mode" (to be woken by CAN activity) is under 30mA. There is another possible supply mode (connected to KL15, not directly to Bat via VBAT2) where consumption in ignition signal mode, when ignition is off, is 0mA (approximately 0,1 µA).
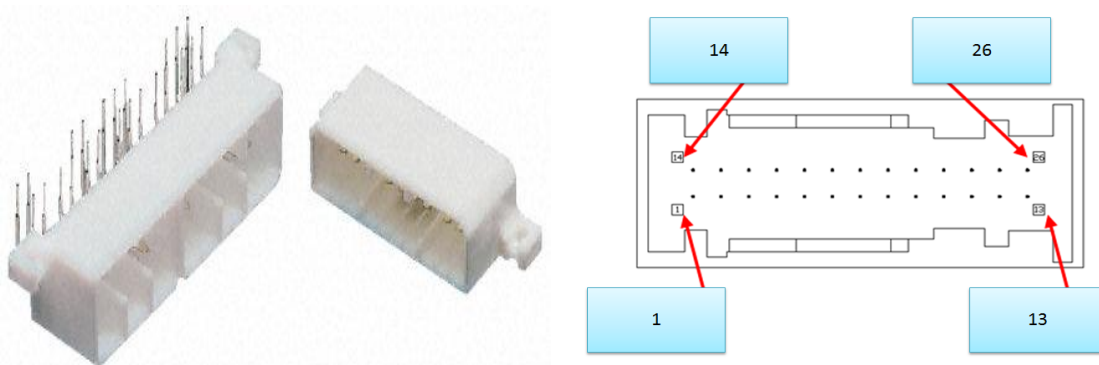
Figure 8: Tyco Connector.

SW Upgrade for Datalogger v2.0 (CTAG Datalogger II) to fulfil euroFOT requirements

- CAN processor: Real-time operating system with enhanced CAN driver.

- New Application processor based on Linux operating system, with applications for CAN & GPS logging.

- GPRS connection to internet that allows real time data synchronization and real-time monitoring/tracking with CTAG "Server tool".

- Diagnosis capabilities added. As a first approach, the diagnostic capabilities were hard coded in firmware. In particular, a flow control according to ISO 15765-2 (Diagnostic TP layer) standard was implemented for the transport layer. It allowed establishing the necessary flow control to perform DIAG ON CAN with several ECUs at the same time. (Currently, by python scripting, the user should be able to perform DIAG ON CAN functionalities according to current standards).

- Synchronization when engine off.

- Firmware update through SD card or remotely (via GPRS).

- Storage data format open (non-proprietary). The format was documented and sample source code was provided.

- Two operation modes are possible: ignition signal dependant power on / off and CAN activity triggered power on / off.

- Modifications on Server tool, Storage and Upload Procedure. As the German-1 operation centre decided to have a management system implemented on the data server, a solution based on the separation of the CTAG "Server tool" functions was preferred. This automatically initiates the process for data upload, pre-processing and storage on the data base. As a consequence, some of the functions of the server tool were separated in the form of Phyton scripts, which enables the user to integrate and initiate the functions according to the definition of the process-chain in the management system. The changes to reduce energy consumption also affected the storage and upload procedure. Therefore, the server tool was adapted to enable both data collection and upload of stored data on the SD card simultaneously.

- Script possibilities implemented (Phyton based). The deactivation process of the data logger after the engine is switched off was additionally integrated into the available Phyton scripts.

- Modifications on Configuration tool: The modification of the energy consumption also affected the configuration software. Hereby the function of using the KL 15 possibility was implemented and integrated as an operation mode in the configuration software.

- Configuration file stored in Flash memory: in case of SD card corruptions, the data logger could only be approached after replacing the new SD card. Hence, it was decided to store the configuration file in the internal Flash of the data logger, in order to have the data logger still accessible for diagnostic operations.

In addition to these, some extra modifications were done to achieve the currents status. Some of them triggered by German-1 OC or French VMC and some more decided by CTAG in order to improve and optimize CTAG Datalogger II capabilities, gaining more reliability:

- Serial port logging capabilities for video synchronization

- SPI connection availability for future upgrades

- Supervisor included in order to avoid hang ups due to some ignition voltage signal drops.

- Possibility to work as HTTP, RSYNC and SSH server

- Reliability improvements after pilot testing.

The final connectors in CTAG Datalogger II are:

- MiniUSB connector for the usage with PC

- FME connector for GPS antenna

- SMA connector for GPRS antenna

- SD card socket

- SIM card socket

The layout of all these connectors mounted in the data logger is presented in Figure 9.
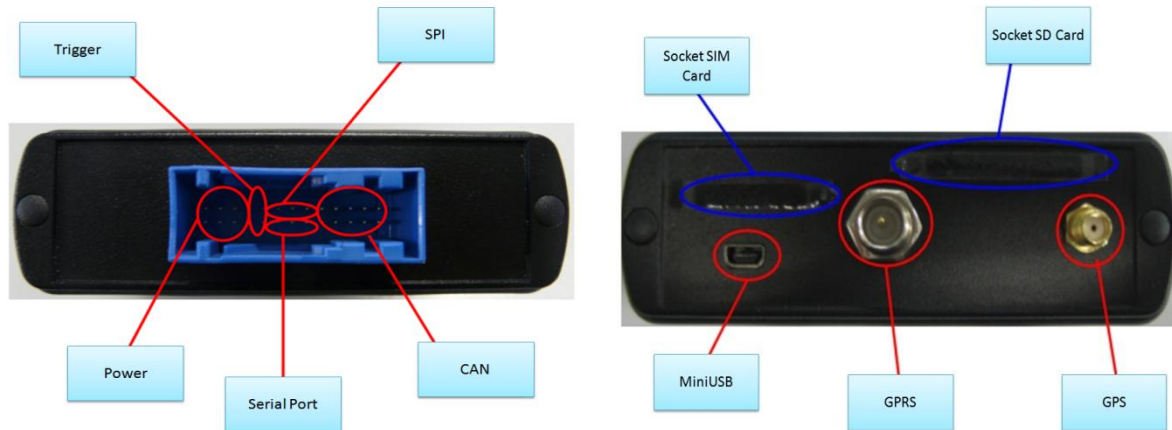
Figure 9: Connectors of CTAG's Datalogger II.

The current operation modes of the DataLogger 2 are (see D3.1):

- MSC (Mass Storage Device): When USB is detected. Both file transfer and data logger configuration is possible.

- Logger: Steady-state operation;

- Idle Mode: When the data logger is powered by VBAT2 terminal (pin 3) and detects CAN inactivity for a period of time configured by the user, it goes into idle mode. The data logger will come back to ``Normal´´ mode again if CAN activity or trigger is detected or if a ``Wake up Alarm´´ event is fired. This mode is not used for the FOT, i.e. no connection via VBAT2;

- Power down: data logger will switch off, if ignition signal (KL15 pin 2) is off, it is powered by VBAT terminal (pin 1) and KL15 is used. The data logger will return to power ON when KL15 is ON.

The final specifications achieved in the CTAG Datalogger II can be summarized as follows:

- The datalogger is 110 (W) x 36 (D) x108 (H) mm and weights only 274 g (excluding cabling). Thus, it is possible to install it in several places in the vehicle (glove-box compartment, trunk space, under driver or passenger seat, etc), allowing a naturalistic experiment approach.

- No fan (no cooling device). Noiseless device.

- Storage temperature ranges from -40ºC to +85ºC. Operating temperature is located between -30ºC and +70ºC, fulfilling euroFOT requirements regarding thermal stress.

- Power consumption limited to 23 mA in stand-by mode (CAN activity wake-up mode). Powering on by ignition signal means no consumption. In this case, shut down time is configurable. If the battery voltage is below, or drops under a certain threshold, the system shuts down.

- Filtering options have been added to the configuration tool. Using standard dbc files, the device can be configured for extracting CAN signals from CAN messages.

- Triggering conditions for data filtering can be programmed by callback functions (Python Scripting). By data compression, also implemented, the transmitted amount of data was reduced to 1/3 of the logged data. In case of both German-1 OC and French VMC the data stored in the device are not readable, for security reasons, before applying specific euroFOT dedicated conversion tools.

- Data in euroFOT is transmitted via GPRS each certain time (configurable). It could also be transmitted after a complete trip cycle, thus preventing overwriting. If the memory card is full and it is not possible to transmit data, the Datalogger stops logging and saves the last data. It does not overwrite data.

- Real Time Clock synchronized with GPS time. The Datalogger timestamps and synchronizes frames from video-logger if required. The timestamp resolution is around 1 ms in CAN data. The maximum latency of a single CAN message is around 40 microseconds. Sample rates for CAN data are configurable (from 0 to 2^{32} ms)

- Storage data format is open (non proprietary) for euroFOT purposes.

- Output data format is .asc compatible with most commonly used CAN analysis network tools. The Datalogger provides code examples allowing data access. The Datalogger has several standard servers (http, rsync, ssh) allowing data uploading, real time monitoring and remote configuration.

- The Datalogger is capable to log up to 4 CAN channels, including extended CAN. The Datalogger is able to send out CAN frames by phyton scripting, and to perform Diagnosis functionality also by programming specific scripts in Phyton.

- Other functionalities such as configurable LED indicators, SDHC capability, USB interface or remote configuration / firmware upgrade have also been implemented and successfully tested.

2.2.1.4 Tests and quality check performed

Table 5: The different tests performed on CTAG Datalogger II.

Test type	Comments
PC tests	Test in laboratory simulating driving cycles were carried out in some cases even running for 48 h (non-stop).
Environmental tests	The reference standard for the product validation was ISO 16750 "Road vehicles – Environmental conditions and testing for electrical and electronic equipment –", in its different parts:
	• Electrical tests: Supply voltage range verification (9-18 V for 12 V nominal, 19-30 V for 24 V nominal), overvoltage immunity (20 V for 12 V nominal, 34 V for 24 V nominal) and reverse voltage immunity (14 V for 12 V nominal, 28 V for 24 V nominal), variations and discontinuities in supply voltage and superimposed alternating voltage.
	• Environmental: storage at 85ºC, operating temperature range (-30ºC, 70ºC), thermal shock resistance (-30 +70ºC), Humid heat (up to 97% HR). Tests have been performed with no remarkable physical effect on the device, whose behaviour was considered as acceptable.
	• Mechanical: Vibration tests following automotive profiles, mechanical shock resistance (500 m/s2 pulses)
	• Chemical: Chemical tests according to 16750-part 5

Test type	Comments
	• Dust and fluid tests (IP50) carried out successfully. Achieved IP50 rating (limited dust protection, no protection from liquid) for dust and fluid tests."
EMC tests	EMC requirements were met (e-marking). E-Marked and M.A.N. EMC requirements (most restrictive ones) compliant.
Real driving tests	About 10.000 Km were done by CTAG. Intensive drives with several vehicles (passenger cars and trucks) were carried out, analyzing the performance of the DAS under those circumstances, including some vehicles with video logging and additional external sensors.

Web-based monitoring tool

A tool (Figure 10) to monitor the status of the vehicles and the DAS is also built, as part of the data management framework used in German-1 OC and French VMC (described in Section 3.2.1). The tool is called "Fleet Manager". Besides an overview of all managed DAS on the server site and their current operation state, additional processes of the framework have been integrated and can be directly accesses by using this GUI. By selection of one DAS detailed state information can be displayed, e.g. the current firmware version on the DAS, the content and state of the FLASH storage, the current GPS position etc. Each DAS needs to be registered at the server before starting uploading data to the server. When a possible issue is identified or attention is needed, an email is sent to the quality management team.

2.2.1.5 Experiences and lessons learned

In the selection and adaptation of a suitable DAS for German-1 OC, some interesting experiences have been obtained. These experiences (for CAN-only loggers) can be summarised as follows:

- Reliability is a key factor in the DAS selection and integration. It should be considered that the system will be running during 12 months in vehicles owned by non-professional drivers. These drivers are real customers from euroFOT partners (OEMs), so no further maintenance calls are desirable.

- GPRS connectivity should be taken into account. As the vehicles can be driven around different European countries and data servers are located in Germany, network compatibility should be considered from the very beginning. In this sense, specific negotiations with service providers (GPRS connection) are to be arranged (e.g., flat rate, etc.). Roaming, connection fade outs and other data-missing scenarios should be considered during the development and pilot testing. The different configuration access parameters within the different European GPRS network providers can cause some configuration problems.
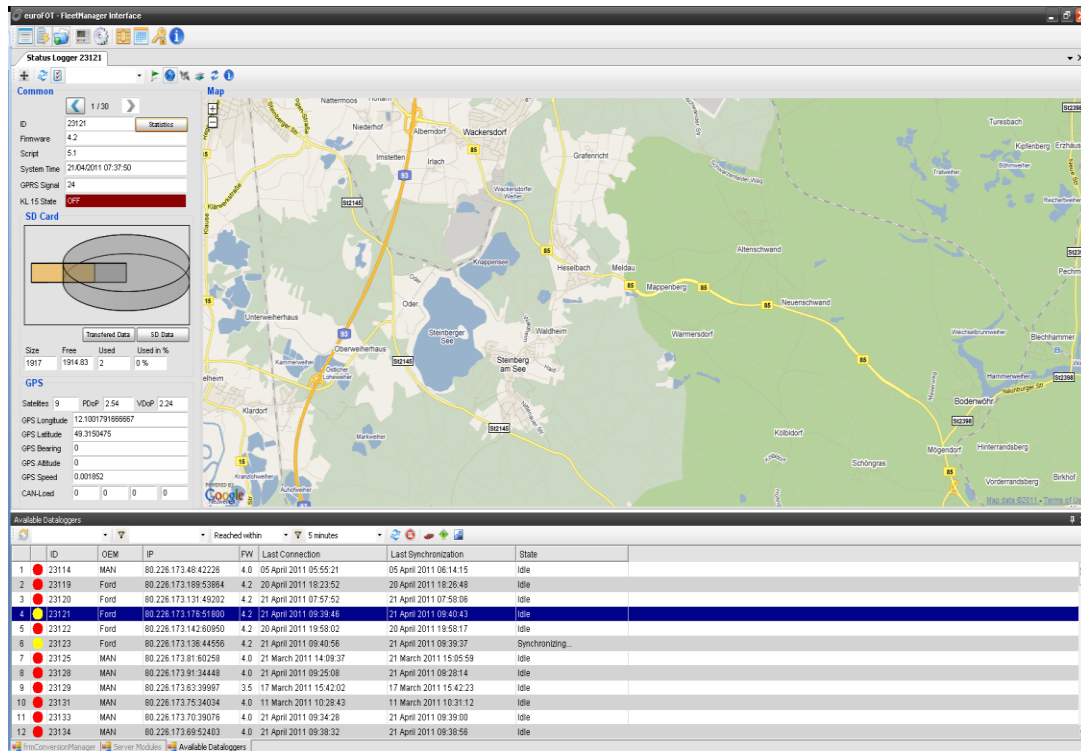
Figure 10: Web-based monitoring tool used by German-1 OC and French VMC.

- Power Management. In order not to drain the battery, a reliable power management should be performed. In this sense, it was necessary to adapt the logging system to work only with KL15, being suitable for series vehicles. In the former version of the CAN DAS, this possibility was not considered, as it was prepared for R&D activities.

- An appropriate management of the data transmission is also important: when to send the collected information via GPRS should be defined: when the vehicle is running and collecting data? At the end of a trip?

- Data Size. As the data to collect in the framework of euroFOT is limited, there is no use in recording and storing all data circulating in the CAN network of the vehicles. An appropriate data filtering (through SP2 and SP4 information) is compulsory. This is a key factor to reduce storage and transmission requirements.

- SD storage. As a back-up in case of transmission failure, and in case GPRS is not able to cope with the necessary bandwidth for the real-time data transmission, an SD Card storage is necessary. As there is the risk to write and overwrite it continuously, a reliable and high-grade card is requested.

- Flexibility. EuroFOT comes up with different requirements in terms of data acquisition systems, specifically for the OEMs (DAS should withstand their different reliability tests and specifications) and in terms of collected data. It is important, then, to integrate open protocols so that the DAS can be customized per OEM and per application. This is done through python scripting.

- Finally, due to all the above mentioned characteristics, it has been revealed that a success factor for the project was the inclusion of the DAS provider among the project partners, so that the system can be updated and upgraded to euroFOT final requirements. This includes future troubleshooting.

2.2.2   French VMC

2.2.2.1 Initial condition

Loggers Vehicles	CAN Only	CAN + Extra sensor(s)	CAN + Video + Extra sensor(s)
Renault Cars	-	**35**	**5**

From the beginning of the project, French partners wanted to have two levels of instrumentation, for two different types of cars. The first level, initially called "CAN only" would have consisted of a single CAN datalogger used in participants' personal cars. The second level, initially called "CAN, Video and Extra Sensor(s)" required CAN logging, video logging, and a radar sensor to monitor headway. This instrumentation level would be used in cars which would be lent to participants for some periods. The reasoning behind those choices was that video was mandatory to validate classification algorithms during the analysis, but that once robust enough, those algorithms could be used with confidence on vehicles with simpler instrumentation where this visual validation would not be possible. It also resulted from the fact that fitting a whole fleet with a complete instrumentation would be much too expensive.

However, during the design phase of the project, it appeared that studying a longitudinal regulation support system without monitoring headway on all vehicles would be quite limiting. It also appeared for some hypothesis that measuring drivers' glance and vehicles' lateral position in lane would be really useful. Therefore, it has been decided that, if that proved to be feasible within budget constraint, all vehicles would be fitted with a headway measuring device, and that the five vehicles with higher instrumentation level would be equipped with lateral position measuring device and an eye tracker.

Generally, the French VMC requirements are similar to the requirements of German-1 OC (see Section 2.2.1.1). However, French VMC also has some additional or specific requirements. The main additional or specific French VMC requirements are:

- The two levels of instrumentation should be compatible (i.e., the richer instrumentation should be an enhanced version of the lower level of instrumentation, relying on the same central data logger).

- All data (CAN or non-CAN data) should be logged into the same data format as CAN-data in the logger (i.e., external sensors that produce CAN signal(s) or those that can be interfaced to CAN with a gateway, are favoured).

- Videologger must have enough capacity to store data for two weeks. This is because of the experimental design in the French VMC; the five cars with richer instrumentation were planned to be rotated among drivers and these cars were planned to be lent in a two-weeks based period to each of the drivers, both during baseline and test phases.

- Videologger must be embeddable, autonomous, and affordable

- Videologger must generate efficient and easy to use files (i.e. use a modern, standard codec).

- Videologger must allow synchronization with CAN-logger with a better than an image (40ms) precision.

- Videologger must have at least 4 video channels.

2.2.2.2 Chosen solution

Several options were considered, including the eventuality to use different, dedicated dataloggers for each instrumentation level.

However, CEESAR finally designed a **modular** system, based around a single datalogger model: a CTAG Datalogger II (same as the DAS used by German-1 OC, see Section 2.2.1.2). This modular system can be as simple as the "CAN only" (and GPS) datalogger, but can also support radar, videologging, eyetracking, and lateral position measurement.

This modular approach has two main advantages:

- **Flexibility:** the level of instrumentation can be adapted car by car

- **Compatibility:** All different levels of instrumentation produce comparable data. Hence, there's no need to harmonise the datasets or convert files produced by different data acquisition systems.

It is also quite cost effective: in its most complete implementation, it is not significantly more expensive than more integrated data acquisition systems which implement the same functionalities (but which can't exist in a simpler version).

It finally proved feasible, after screening of a lot of suppliers and long negotiations, to, within budget constraints, reach both goals:

- Fitting the complete fleet with radars

- Equipping all five "high level" instrumentation cars with video logging, lane tracking, and eyetracking devices.

Table 6: The part list used for the two levels of instrumentations used in the French VMC.

Item	Low Level	High Level
CTAG Datalogger II	●	●
Max 4 CAN Channels	2 channels used	4 channels used
GPS	●	●
GPRS data transfer	●	*(not used : manual transfer)*
TRW AC20 radar	●	●
(not part of standard vehicle equipment)		
VideoLogger		●
(custom made for CEESAR, H.264)		
Cameras		4
(B&W, SuperHAD Exview)		
Mobileye AWS		●
(added, with special firmware)		
Smarteye Eye tracker		●

The components of those two levels of implementation are described in the rest of this section.

Low level instrumentation
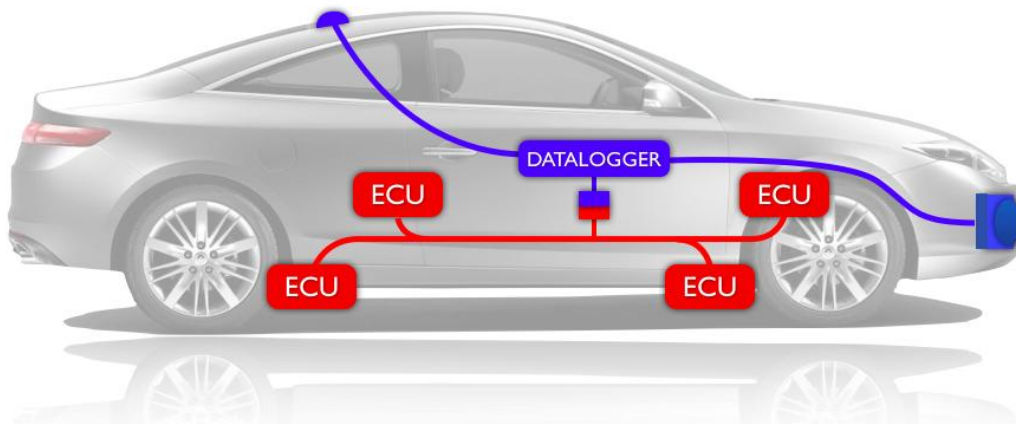
Figure 11: Low level instrumentation at French VMC.

In addition to the CTAG Datalogger II, the low instrumentation level consists of an ACC radar. This is a standard AC20 radar from TRW (Figure 12), including a gyrometer (allowing function on a vehicle without ESP), and with a software which, instead of controlling ACC and/or FCW, just provides the characteristics (distance, speed, azimuth, etc) of the vehicles in front of the car.

Figure 12: AC 20 radar.

Integrating the radar in the DAS required the development of a callback mechanism to forward some information (mainly vehicle speed) from the vehicle to the radar.

High level instrumentation
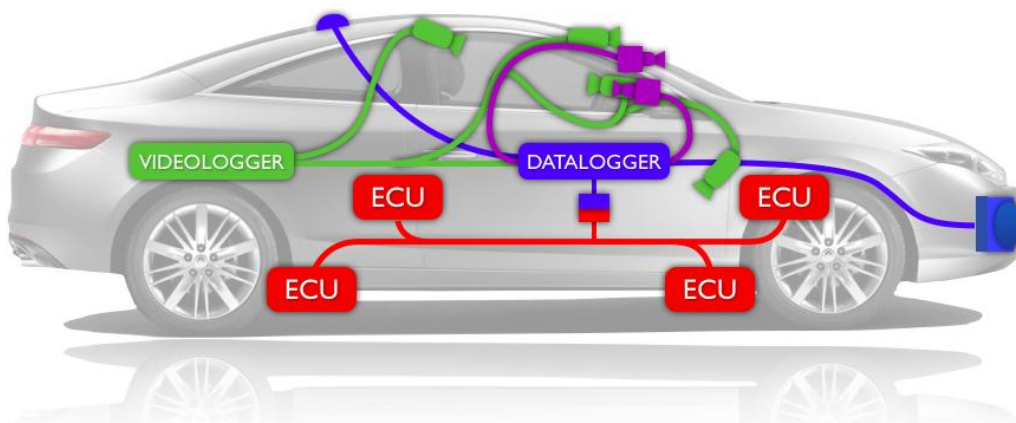
Figure 13: High level instrumentation at French VMC.

The high level instrumentation starts from the low level instrumentation, and adds the following elements:

- Videologging system (videologger and cameras)
- Lane tracking system (Mobileye)
- Eye tracker (Smarteye)

2.2.2.3 Final implementation to fulfil euroFOT requirements

CAN-only system

Same implementation as in German-1 OC, see Section 2.2.1.3.

The only difference in CTAG Datalogger II that had to be implemented specifically for French VMC was the synchronization mechanism which consists on some frames received from the videologger by the Datalogger serial port. The Datalogger acknowledges the reception and correction of the frames and saves the information of the frames within the log so that the video frames can have a time-stamp with the same reference as the CAN data one.

Videologging system

The videologging system (Figure 14) consists of a videologger and videocameras.

Figure 14: Video logging system at French VMC.

It was initially thought that off-the-shelf video surveillance systems could fulfil all those needs. However, after an extensive screening process, it appeared that no such system existed. Some systems were autonomous, embeddable, and cheap, but required unsuitable and specific cameras, and used proprietary file formats or poor compression methods (such as

MJPEG). None of the systems reviewed allowed precise synchronization through RS232, which was the only available port on the CTAG datalogger.

It was therefore decided that CEESAR would develop its own video logger. The result is based on Axis Communications Ethernet video loggers, which have the following advantages:

- They use the very efficient H.264 compression

- They accept composite video inputs, which allows using any standard analogical camera

- Their network capability allows using any combination of encoders on the same network, to use as many video channels as needed.

Currently, a single Q7404 is used, which allows for the 4 channels which were required.

Video logging itself is ensured by a NEXCOM VTC3300 (VTC 3300, 2007) embedded PC, running Windows XP, and an application programmed using Axis Communications' SDK.

The cameras which are used are standard analogical video surveillance cameras with the following characteristics:

- 1/3" CCD Sony Black & White, SuperHAD Exview Image Sensor

- Interchangeable lens (f/2 aperture)

- 3cm wide enclosure embedding all electronics

Those were chosen after an extensive review for their great efficiency vs. price ratio. Different cameras were tested in various driving and illumination conditions: colour cameras were dismissed not only because colour isn't necessary for analysis and requires larger storage, but also because even last generation cameras didn't show the same sensitivity and dynamic range as the black and white cameras used. Specific automotive cameras were also considered and tried for the front scene, but their interfacing was not standard (connectivity and power supply), they were more expensive, and the benefits they brought in high dynamic range compensation were not sufficient to justify the effort.

Feedback from pilot testing showed that with typical driving scenarios, video compression on 4 channels was sufficiently efficient to allow recording nearly 200 hours of video on the single embedded 80 Gb automotive hard drive. This proved much more than enough for euroFOT requirements. Video records are transferred to the storage server by network connection, when vehicles are at CEESAR, using a simple windows batch script. As video files are directly recorded with the very efficient H.264 codec, no further re-encoding is necessary, and they are directly used as-is by the data analysis tool.

Synchronisation information being transmitted regularly during recording from the videologger to the datalogger, they are embedded inside the CAN & GPS data acquisition files. Compensating for latencies in video encoding and network transfer (between framegrabber and embedded PC), very precise synchronization was obtained (i.e. the delay if any is much smaller than a frame duration).

Lane tracking system

The lane tracking system used is a Mobileye AWS2000 (an aftermarket Lane Departure Warning and Forward Collision Warning, see Figure 15) (MOBILEYE AWS-2000, 2008). It has a modified firmware which allows retrieving by CAN the values measured by computer analysis of the front scene. Currently, only the position on Lane is saved (position of other cars can be retrieved as well, but radar provides better measurement in at higher distances).

Figure 15: Mobileye AWS2000.

Eyetracking system

The eyetreacker used is an Antisleep EHT from Smarteye. It is a single camera eye tracker (camera, computer and IR illumination), placed in the centre of the dashboard.

Integration

The integration of equipments is described in D2.2: Report of the results of adaptation, in-vehicle implementations and piloting and the installation guide which is an annex of D2.2 (Saint Pierre, Kessler, & Malta, 2011).

Figure 16 shows the visible elements of the instrumentation. In the low level instrumentation, only the radar can be seen inside the front bumper.

Figure 16: French VMC - Placement of extra sensors and cameras.

2.2.2.4 Tests and quality check performed

Most of the tests and quality check performed are already described in Section 2.2.1.4. Some other tests performed on videologging are listed in Table 7.

Table 7: The different tests performed on videologging of French VMC.

Test type	Comments
PC tests	The potential and suitability of Nexcom VTC3300 as videologger were tested.
Video quality	Many tests were done to check the quality of the video from different cameras tested in different driving with different illumination conditions.
Synchronisation test	The delay in synchronisation between video frames and CAN-data was measured. Necessary changes were made until acceptable synchronisation is achieved.
Real driving tests	Was done during piloting

Web-based monitoring tool

Same as the one described in Section 2.2.1.4 on page 20.

2.2.2.5 Experiences and lessons learned

Simpler and cheaper cameras could be more suitable for FOT. Standard, although rather high end (SuperHAD Exview sensor), analogical black and white video surveillance cameras were found sufficient in the French VMC. Colour cameras require larger storage and even the last generation cameras did not show the same sensitivity and dynamic range as the black and white cameras used. The specific automotive cameras have higher dynamic range, but they do not use standard connectivity and power supply, and were more expensive than the black and white one.

2.2.3 Swedish VMC

2.2.3.1 Initial condition

Vehicles \ Loggers	CAN Only	CAN + Extra sensor(s)	CAN + Video + Extra sensor(s)
Volvo Cars	-	-	**100**
Volvo Trucks	-	-	**30**

The partners in the euroFOT-Swedish VMC are Chalmers, Volvo and VCC. Within euroFOT, data management tasks in the Swedish VMC were especially challenging due to the complexity of the data acquisition systems and the number of safety functions tested. Furthermore, fleet handling was also comparatively arduous because 15 of the participating Volvo trucks are based in the UK and 15 in The Netherlands; many of these trucks often be driven outside their home country. Thus, special management and maintenance strategies were needed. All of the participating VCC cars are based in Sweden.

While waiting for the list of measures from SP4, SP3 partners inside Swedish VMC started gathering requirements based on partners' experiences from previous smaller FOT studies run in Sweden. The result from this step was a preliminary requirements specification that was used as a basis for request for information (RFI). This RFI was used to identify companies that are interested in participating in the project as suppliers of equipment and/or services for the data acquisition system.

At the same time when preparing the RFI, a survey was conducted to get an initial list of potential suppliers. The RFI was then sent out to potential suppliers of the main DAS system and sensing equipment as well as companies that have been identified as possible system integrators during November-December 2008. Since then to July 2009, the preliminary requirements specification and the list of potential suppliers were constantly being revised as more information became available.

After receiving the list of measures from SP4, SP3 partners worked together with SP4 to identify the necessary signals for the measures and particularly the two OEMs in the Swedish VMC identified where to retrieve those signals (i.e., from which vehicle network the signals can be retrieved). The information that was gathered from this activity was used to revise the preliminary requirements specification. The rest of this section discusses the key elements of the technical requirements of the Swedish VMC.

Swedish VMC technical requirements

System requirements

1. It must be possible to equip the DAS with up to four (4) CAN buses.

2. The DAS should be able to store all data from four CAN buses.

3. It should be possible to configure the DAS to store data in the following ways for all buses (CAN, LIN and J1708):

- All frames from a bus.

- Specific frames from a bus (i.e. filtering on frame identifier).

- Specific signals from a bus (i.e. filtering + decoding).

4. If the DAS is configured to decode specific signals from a bus, there must be a secure method (e.g. encryption) to protect the configuration file from being read by any unauthorized person.

System requirements

5. The DAS must have functionality to send out CAN frames by routing frames from one bus to another.

6. It must be possible to equip the DAS with, and collect video data from, at least four (4) video cameras. Desirably, the system should be able to handle six (6) cameras.

7. The DAS must be capable of combining images from all cameras into a single image, 640x480 (or similar).

8. The DAS must be capable of storing video data at a rate of 10 Hz or more.

9. The DAS must compress all data from all sensors (all CAN buses, all cameras etc.) to a size below 1 GB per hour.

10. It must be possible to equip the DAS with, and collect data from, two (2) LIN buses or

two (2) digital I/O:s.

11. It must be possible to equip the DAS with, and collect data from, one (1) J1708 bus.

12. The DAS must have a GPRS or 3G modem.

13. The modem must comply with European cell phone standards.

14. The DAS must have a GPS receiver. GPS data (coordinates, UTC time, speed, etc) must be possible to be stored just as any other data. GPS data must be stored at a rate of at least 1 Hz.

15. The DAS must have an option to be equipped with, and collect data from, either a forward or a rearward looking RADAR delivered by the DAS supplier.

17. The supplier must provide information about the average latency and jitter for each signal source.

18. The synchronization inaccuracy between different signal sources must not be larger than 100 ms. (When post-processing the data based on the latency and jitter figures).

19. The DAS must autonomously log data from all signal sources (CAN, video, LIN or digital I/O, J1708, GPS, etc). There must not be any driver interaction needed at all.

20. When the ignition is turned on, the DAS must automatically boot up and start logging.

21. When the ignition is turned off, the DAS must automatically stop logging and turn off. The shut-down time should be as short as possible.

22. The operation of the DAS must be controlled by a configuration file.

23. In the configuration file, it must be possible to specify parameters for each signal source where applicable. Such parameters might include, but not be limited to:
- CAN baud rate.
- Which CAN frames or signals to collect.
- Video frame rate.
- Video quality.

24. It must be possible for our personnel to edit the configuration file and to update the DAS with the new file.

25. The DAS must have functionality to automatically connect to a server via GPRS/3G and check if there is any updated software or configuration file. If there is, the new software and/or configuration file should be downloaded.

26. The DAS must have functionality to automatically connect to a server via GPRS/3G (e.g. once per trip) and upload a status report containing the current status of the device. The status report should include, but not be limited to:
- DAS status.
- Remaining hard disc drive space or low disc space warning (configurable).
- Detected errors.
- Desirably quality measures for some signals in order to assess the quality of sensors, e.g. eye tracker.

System requirements

- Desirable a measure of the hard disk drive health.

27. The DAS must have functionality to minimize or turn off data transfers over GPRS/3G when roaming.

28. The DAS must have a storage capacity for collected data of at least 80 GB.

29. It must be possible to extend the storage space, either by changing storage media (e.g. using a larger hard disc) or by adding an external storage media on USB. It is understandable that the storage media larger than 80 GB could not operate in the full operating temperature range.

30. The process of exchanging storage media must be easy and fast (< 5 minutes). This means that there must be a hard disc drive bay or that the hard disc drive is external to the main DAS unit.

31. The DAS must be able to encrypt data on the storage media.

32. The data format of the collected data must be available to euroFOT partners in Sweden for development of tools for reading the data files.

33. There must be a mechanism (e.g. watchdog) that guarantees that the DAS always is shut down when the ignition is turned off.

Hardware requirements

1. It must be possible to order the DAS with the possibility to send out CAN frames physically disabled on three of the four buses.

2. The DAS must support being powered from both +12V (cars) and +24V (trucks) electrical systems, also during engine crank. This implies that applicable automotive standards like ISO-7637 should be met. The supplier must guarantee that the DAS will work properly when powered by the electrical system in a car or truck.

3. The operating temperature range of the DAS must be -20°C to +55°C or better.

4. The DAS must not consume more than 10 mA at 12V in off mode.

5. The DAS must shut down automatically if the battery voltage of the vehicle falls below a certain limit. The threshold level must be configurable in software or in the configuration file for the DAS.

6. The supplier must provide all necessary mounting equipment and cabling necessary for a full installation. This must be provided for three different car models (preliminary Volvo XC70, XC60 and V50) and two different truck models (preliminary Volvo FH and FM).

7. The DAS must have a method for the driver to turn off and on the video recording of the vehicle's interior, desirably while the logger should continue to collect all other data and video. There must be an indicator telling if the video is on or off, or that the driver can be assured that the video is turned off in another way.

8. The main DAS unit must fit into:
a) A cylinder that is 630 mm in diameter and that has a height of 140 mm (for cars).
b) A box that is 320 mm x 150 mm x 400 mm (for trucks).

9. Cameras should be "as small as possible". It is expected that three of the cameras will be mounted by the rear looking mirror and they should obscure the driver view as little as possible.

10. All equipment must be CE marked and also pass a number of additional tests or fulfil a number of OEM's EMC requirements.

2.2.3.2 Chosen solution

Based on the requirements, two different options were considered:

1. Develop a system based on PC technology and commercial off-the-shelf components (COTS);

2. Develop or buy a dedicated non-PC based solution that is developed for FOT purpose.

While the option to buy a dedicated non-PC based solution seems very easy and attractive, our market survey and the result of our request for information (RFI) in 2008 suggested that there was no readily available commercial system on the market fulfilling all of the requirements specification and the budget constraints of the Swedish VMC. Indeed, there were very few choices of full-logging system available on the market, not to mention those designed for large scale field operational test. Therefore, it was decided to also focus on a new custom-made DAS (PC-based solution) designed to fulfil euroFOT-Swedish VMC requirements specification. For this, a further extensive market survey on the DAS components was conducted in early 2009. Part of this market survey was done via internet survey. For this activity, previous experiences from earlier FOT projects were very advantageous.

Request for tender (RFT) documents (for full DAS, Vehicle PC, eye tracker and vehicle communication interface) were prepared and sent out in July 2009. No answer was received for the RFT for full DAS. Thus, there is only one solution; that is to build a new custom made DAS based on PC technology and commercial off-the-shelf components.

Relation between the chosen solutions to previous SAFER-owned data acquisition systems

The euroFOT partners in the Swedish VMC are partners of SAFER, a Vehicle and Traffic Safety Centre at Chalmers. The SAFER partners (including the euroFOT partners in the Swedish VMC) have previously involved in other FOT studies in Sweden. In those studies, the SAFER partners have used two different data acquisition systems. The designs of these two systems can be used by the Swedish VMC partners for this study. However, due to more complicated and restricted requirements for euroFOT, both systems would need further development to fulfil the euroFOT-Swedish VMC requirements specification. For example, EMC requirements are not yet fulfilled by the previous data acquisition systems.

EMC requirements have never been a strict requirement that had to be fulfilled by the DAS in the previous FOT projects. This was because the vehicles used in the previous projects were all classified as test vehicles. Therefore, non-EMC classified equipment could be installed.

When compared to the previous SAFER-owned data acquisition systems, this new system has a different hardware platform but was built using some of the designs, ideas, and software pieces from the previous SAFER-owned data acquisition systems.

2.2.3.3 Final implementation to fulfil euroFOT requirements

The parts that are used to make up the DAS used in the Swedish VMC (SAFER-euroFOT DAS) are listed in Table 8. Figure 17 shows the placement of the DAS in the spare wheel well of a Volvo XC70 car and the placement of the cameras.

Table 8: The components used in the SAFER-euroFOT DAS.

Item	Comment
PC - Nexcom VTC6100	Intel Atom 1.6 GHz. The Nexcom VTC6100 (VTC6100, 2010) is a rugged PC developed for automotive applications. It has regulated +5V and +12V outputs, 1A each. This made it possible to power all equipment without

Item	Comment
	any additional power supply (except for the eye tracker). Furthermore the VTC6100 has an elaborate power supply with input for both KL15 and KL30 directly from the vehicle.
GPRS/3G modem	PCI, internal to the PC.
GPS receiver	Internal to the PC.
Video grabber	Ampltd MPEG4000XLP (MPEG4000XLP, 2011). MPEG4 HW compression, PCI. The video grabber is fitted internally to the PC. It has four CVBS video inputs. The SW API generates a callback for each coded frame (MPEG4).
Compact Flash	Industrial specification. Transcend 16 GB, industrial grade. The Compact Flash (CF) memory is used both for the Windows XP Embedded installation and for primary data storage during operation.
Hard disk drive	Seagate FreeAgent Go 500 GB. USB, external to the PC. This HDD has two features that makes it suitable for this application: • It inhibits spin-up if temperature is below 0° C. Spinning up a HDD at low temperatures can damage the HDD. • It spins down after several minutes of inactivity. Data from the previous trip is copied from the Compact Flash memory to the HDD at the beginning of each trip, thereafter the HDD spins down. It is believed that HDDs will last longer if not operated during driving.
CAN interface (2)	Kvaser USB CAN Professional (Kvaser USBcan Professional, 2011). USB, external to the PC.
BLIS interface (2)	Cars only. Custom made by VCC.
J1587 to CAN interface	Trucks only.
Accelerometer - Bosch	It is a two-axis accelerometer with an additional yaw-rate sensor.
Relay	To minimize power consumption in off mode. Cars only.
Cameras (4)	
IR illuminators (2)	
EMI filter + fuse	For eye tracker power supply.
Switches (2)	For driver to turn off cameras. Trucks only. Volvo standard.
Combined GPRS/3G and GPS antenna.	
Eye tracker	SmartEye embedded (Embedded AntiSleep, 2011). Ethernet and CAN.
Cable harnesses	Cable harnesses were manufactured by VCC. One model for cars and one model for trucks. The same cable harness is used for both the left-hand controlled trucks (Dutch fleet) and for the right-hand controlled trucks (UK fleet).
Mounting brackets	Mounting brackets for cameras and the eye tracker were manufactured at VCC.

Figure 17: Swedish VMC - Placement of DAS including extra sensor and cameras.

A number of functions (special features) that improves the performance of this DAS are worth mentioning here.

HDD diagnostics. HDD status is constantly monitored by using a commercial tool called "HDSentinel" (Hard Disk Sentinel - HDD health and temperature monitoring, 2011). This application uses a technique called "SMART" to get information from the HDD. The current status of the HDD is written to the status report that is uploaded once for each trip. Since consumer grade HDDs (i.e. non-automotive standard) are used, it was considered extra important to keep track of the HDD status to get an early warning if the HDD performance starts to decrease (read/write errors etc).

HDD shutdown. The selected HDD has a function that will make it spin down after a few minutes of inactivity. By not having the HDD up and running all the time, it is believed that the HDDs will last longer.

Inbuilt diagnostics. It was early decided to add as many diagnostic functions as possible to the DAS software. This turned out to be helpful both during early tests and during execution of the FOT, both for detecting problems with cable harnesses and for detecting failures within the different components of the DAS.

For each driving cycle (ignition on to ignition off), a status report (XML) is generated in the DAS which is then uploaded wirelessly to a server at the beginning of the next driving cycle (see description of the web-based monitoring tool in Section 2.2.3.4).

The following errors can be detected:

- Missing video signal (e.g. loose video cable or defect hardware).
- Obscured camera.
- Low CAN busload (settable threshold).
- CAN error frames.
- CAN bus hardware buffer overflow.
- Decreased HDD performance (see HDD diagnostics above).
- Missing HDD (e.g. loose USB cable or defect hardware).
- Missing CAN interface (e.g. loose USB cable or defect hardware).
- SW errors.

Furthermore, the status report contains information about:

- HDD temperature
- GPS signal strength
- GPRS signal strength
- Histograms over signal values (configurable)
- Average signal values (configurable)
- Remaining storage space on Compact Flash and HDD.
- Remaining money on SIM card.
- Version number of all software in the DAS.

Silent and galvanically separated CAN bus interfaces. In order to make sure that the logger system cannot affect the vehicle in any way, the CAN interfaces were ordered with the

ability to send disabled. Furthermore, the requirement was that they must be galvanically separated so that there is no current path between the logger PC and the CAN bus.

Configuration. Each DAS is configured with a unique identifier in a file. This identifier is then used to tag the stored data as well as to identify each vehicle on the status web (web-based monitoring tool).

Installation. For cars, the logger unit is installed in the spare wheel well. For trucks the logger unit is installed in the left luggage compartment. Cameras are installed on four different locations in the vehicles; Cars: forward view, driver, feet, rear view; Trucks: forward view, driver, feet, right blind spot (Dutch fleet), left blind spot (UK fleet).

Installation verification. A software application was developed that shows live reading of all signal sources. This gives the installer means of detecting e.g. missing camera signals and CAN bus errors.

GPRS subscription. Cash cards from a Swedish operator are used. These have to be refilled regularly. During roaming conditions (Dutch and UK fleet), the software collects a number of status report before sending them to maximize the use of the minimum fee that applies.

2.2.3.4 Tests and quality check performed

Table 9: The different tests performed on the SAFER-euroFOT DAS.

Test type	Comments
PC test	Of special interest was to test the power management part of the PC since this is a known area of problems in automotive applications. The PC has three power inputs; KL15, KL30 and Gnd. It also has settings for delaying start-up and shut- down times with respect to KL15 changing state.
	The following deviations were found:
	• Higher current consumption in off-mode than specified (17 mA instead of 10 mA). This is only a problem for car installations and was solved by adding an external relay.
	• Unstable operation after resuming from hibernation and sleep. Since these modes were not planned to be used this was considered a minor issue.
Climate chamber tests	A complete system was tested in a climate chamber at different temperatures ranges from -20 to 60 degrees Celsius. The tests were successful. The chipset in the PC has a function to shut down if the internal temperature reaches a certain limit. This function has not been tested since it was considered to be a potentially destructive test and that the available number of PCs was low at the time
EMC tests	A complete system except for the eye tracker was tested at an authorized EMC testing facility (SP Technical Research Institute of Sweden) to verify that it complied with the requirements for e-marking. The tests were successful.
	The eye tracker was tested separately by the supplier and an EMC consultant. The tests were successful.
Real driving test	The DAS was installed in one truck and one car which were driven for some months.
	Software development was carried out in parallel with the in-vehicle pilot tests. Each new function was tested when delivered from the developers.
	Data was extracted regularly and examined both manually and by

Test type	Comments
	software scripts, see below.
	Special attention was put into verifying that the DAS starts up and shut down properly as the ignition is turned on and off. The shut down must work in order not to drain the battery in the vehicle. The PC has an extra functionality (watchdog) to ensure proper shut down also in case of a severe software failure. This function was tested and worked well.
Video quality	During the pilot, video quality was checked by offloading and manually reviewing data as often as possible. An error was found where interlace effects occasionally could be seen in the recorded video. After discussions with the manufacturer of the video grabber card a workaround was implemented in software by changing the way the card was initialized.
	Image quality for the driver view is negatively affected by the limited dynamic range of the selected camera. Extra IR illumination was added to mitigate the effect and showed a slight improvement.
Time stamping performance	The DAS time stamps data by a software mechanism. This means that the time stamping performance is affected by the current workload of the CPU. In order to get a better understanding of the time stamping performance a software tool was developed that outputs text files containing just the time stamps for further analysis in e.g. Excel. The time jitter for CAN and video and the average latency between the signal sources were measured.
CAN quality	The DAS collects and stores all data from four CAN buses. Data is compressed before storage to file. A test environment was created to verify that the logger does not miss any CAN data. A function was implemented in the DAS software to detect and report CAN bus hardware buffer overflow.
Eye tracker tests	A software application was developed to extract eye tracker signals from collected data and to calculate derived values. The eyetracker software has had to be updated several times to address issues found during the tests.
Accelerometer tests	The accelerometer is connected to one of the four CAN buses and is powered from the PC.

Web-based monitoring tool

To accommodate DAS-quality check during FOT, a web-based monitoring tool was also built. This tool has functions for showing the fleet status in four different views:

1. Main page where all vehicles are shown (Figure 18).
2. Vehicle page where all trips for a specific vehicle is shown.
3. Trip page where all diagnostic parameters for a specific trip is shown.
4. Image snapshot page where snapshots from all cameras for a specific trip are shown.

From the main page, red-colour is used to highlight which DAS or vehicle needs attention or has some issues. Information about the CF card and HDD (space left and health status), remaining money on SIM card and last upload from vehicle are also presented in this page. Information about trips and their details (like various measurements to check the different sensors are working fine and trip duration) are shown in vehicle page. Details of detected errors during a trip, version of logger configuration file etc are shown in the trip page.

Figure 18: The main page of Swedish-VMC DAS-vehicle monitoring tool.

2.2.3.5 Experiences and lessons learned

Inbuilt diagnostics with remote status reports is a necessary tool to follow up the status of the DAS systems. Connectors will fail, as do cameras, PCs, IR illuminators, HDDs etc. If vehicles are operated for a long time before data (HDD) is collected, there is a big risk of losing a lot of data before an error is discovered. Note that only the status reports are sent wirelessly to the Swedish VMC centre, the collected data in the HDD is collected manually by OEM personnel (see Figure 36 on page 62).

Failing connectors have been the main source of errors. The reason has been due to human factors, where the personnel assembling and handling connectors have not been skilled/educated/trained enough.

The Compact Flash (CF) cards recommended by the PC supplier (Transcend 32 GB, consumer grade, MLC type) worked well during the pilot but almost immediately broke down when a larger number of vehicles started to run. All cards were replaced with Transcend 16 GB, industrial grade, SLC type cards. The PC has to be opened to replace the CF card and the recommendation is that one should try to find a PC with the CF card (or whatever storage medium that is used) accessible from the outside.

The PC supplier (Rugged Mobile Systems UK) has been of great help. They provided a lot of services so that what was delivered to the Swedish VMC was a complete and tested PC with an installed video grabber. In practice, the Compact Flash problems lead to a significant amount of extra work for the Swedish VMC (opening up all PCs for CF replacement), but in general it is of course recommended to find a supplier that can do as much work as possible.

It is easy to underestimate the work needed for manufacturing cable harnesses and mounting brackets. One complete cable harness took ~8 hours to manufacture.

Since the task to provide suitable DAS for the Swedish VMC turned partly into a development project (there was no equipment (DAS) available on the market that fulfilled the requirements), it would have been convenient to build a complete test rig for emulating the environment. Instead, most full-scale tests were done in real vehicles. For example, the Compact Flash problem would most likely have been detected in an earlier stage if the system had been left continuously running for one or two weeks in a test rig. As it was now, the pilot car was run for several months without failure.

Configuration of each logger with a unique identifier takes a lot of time. One has to connect a monitor and keyboard and log into the system to alter a file. Instead another method would have been better, e.g. by letting the SW application read out the serial number of the PC or something similar.

2.2.4   German-2 OC

The partners involved in German-2 OC are not members of SP3. Their works to select and finally made available DAS systems for FOT-data collection in this OC were made outside of SP3. Here, only a summary of DAS systems used in German-2 OC is presented. Further detailed can be read from Deliverable D2.2 (Saint Pierre, Kessler and Malta 2011) and Deliverable D5.2 (Csepinszky and Hagleitner 2011).

2.2.4.1 Initial condition

Vehicles \ Loggers	CAN Only	CAN + Extra sensor(s)	CAN + Video + Extra sensor(s)
Daimler Cars	-	-	**15**
BMW Cars	-	-	**15**

Within euroFOT, only German-2 OC studied safe human machine interaction (safe HMI) systems. Specifically, they studied the BMW and the Daimler solutions for built-in navigation systems and mobile devices (Z205) manufactured by Harman/Becker. For the purpose of this project, the used mobile devices were modified, in such a way that the use and the handling of the system could be logged on a SD-card in the mobile devices.

The main requirement for the DAS is that the system must be able to collect videos of driver behaviour, in particular videos of driver's face, movements of head and upper body of the driver, movements of arms of the drivers, and the view to the navigation systems, forward view and rearward view.

2.2.4.2 Chosen solution

The two OEMs in this OC have chosen to use their own in-house solutions from the beginning of the project.

BMW developed its own in-house DAS solution.

Daimler contracted MBTech to develop their in-house DAS solution. MBTech was chosen because of their knowledge of Daimler vehicles and that they have the appropriate installation facilities and experiences.

2.2.4.3 Final implementation to fulfil euroFOT requirements

BMW

The components that make up the BMW-DAS are listed in Table 10 and the placement of DAS is shown in Figure 21.

Table 10: Part list for the DAS used by BMW.

Item	Comment
ReadyNAS 2100	Network attached storage (NAS) device used as the main component of the DAS. This component has 4 hard disks, each 1 TB.
Power management	
ZGW (Zentraler Gateway Switch)	Central gateway
Video server ACD2000Q	
UMTS Modem Viprinet 300 including UMTS antenna 2dBi	
8 ports Hub	
Short range radar sensors (2)	
ahe CMC2002HP-1/3.7-PAL	Driver camera
ahe CM-108 ¼' CCD Mini (3)	Cameras to the front, rear view, and centre stack
KISSte (Configurable Intelligent System Control)	
DC-DC Converter	
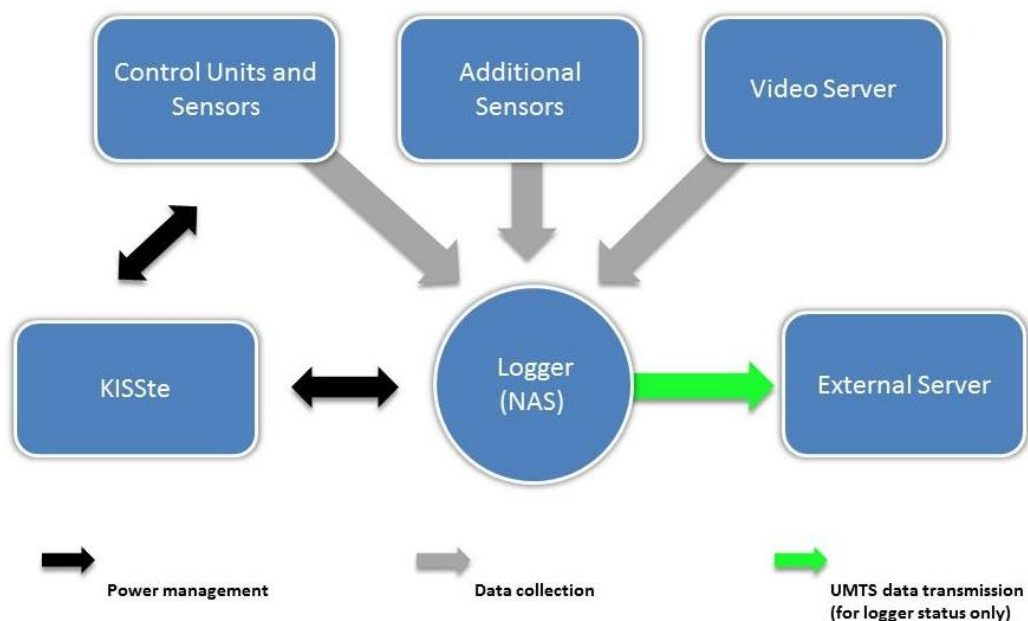
Figure 19: Communication of the data logger components.
This figure is from D2.2.

All DAS components are connected to the vehicle power network via the configurable intelligent system control (KISSte), see Figure 19. This KISSte is also connected to the K-CAN. The functions of the KISSte are:

- to start the DAS components when it recognises a bus wake up signal.

- to shut down all DAS components when it recognises that the bus falls asleep.

- to measure the vehicle battery status. When the SOC value of the battery is too low, the KISSte will prevent the DAS components from starting.

The function of the ZGW is to transfer the signals from the vehicle busses and the extra radar sensors to the NAS, via Ethernet. The data is then stored by the data acquisition software in the NAS. The video data (JPEGs) is transferred to the NAS by the video server via internet. The status of the data logging components is transferred to an external server via UMTS.
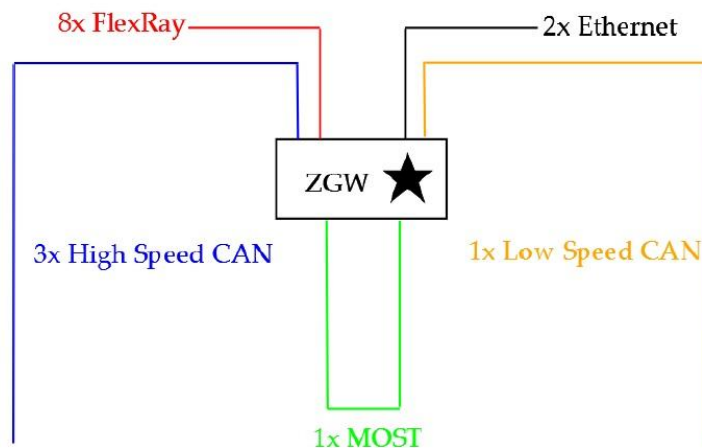
Figure 20: ZGW connections.
This figure is from D2.2.
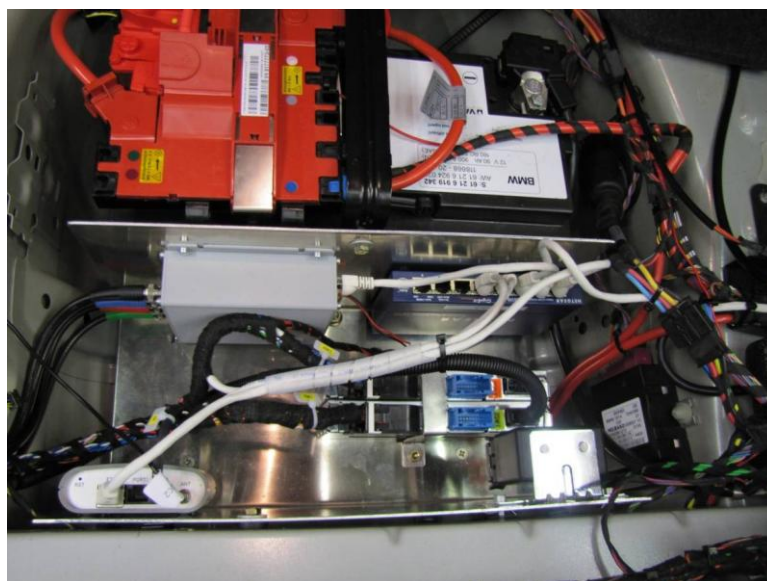
Figure 21: Placement of DAS in a BMW vehicle.
This figure is from D5.2.

Daimler

The DAS logs data from several CAN-buses and cameras. The architecture of the data acquisition system is shown in Figure 22, and the placement of the DAS in a Daimler vehicle is shown in Figure 23.

Hardware
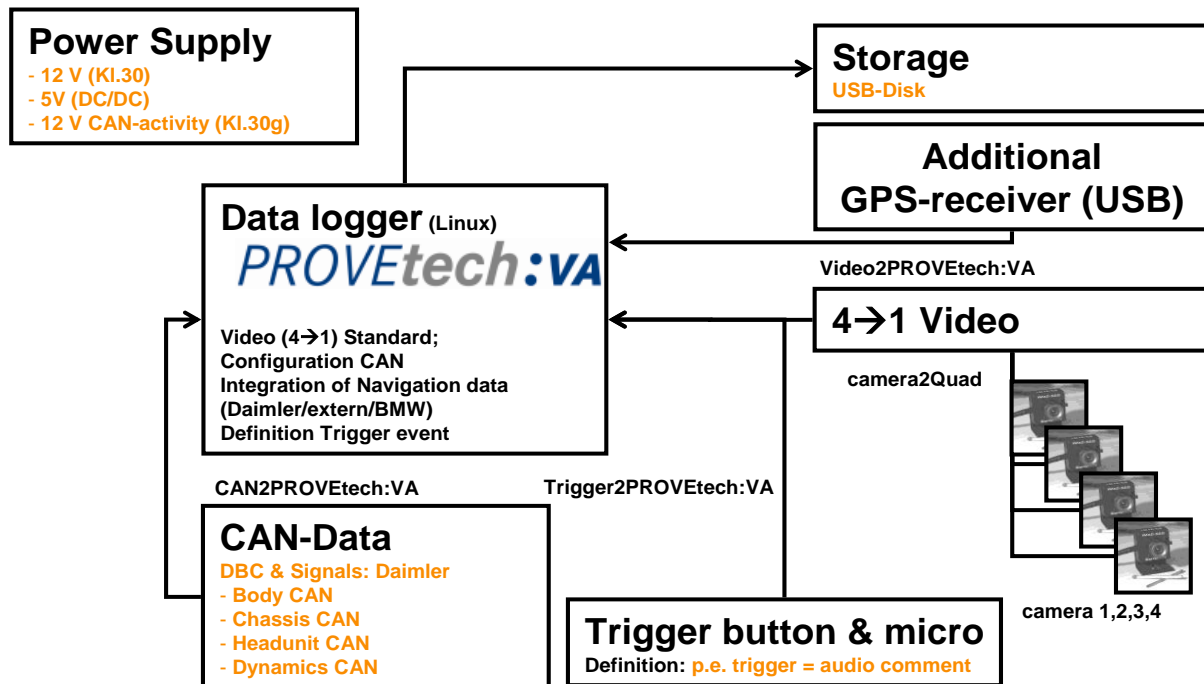
Figure 22: The architecture of DAS used by Daimler.
This figure is from D5.2.

The DAS used by Daimler also includes a trigger box and a microphone (see Figure 23) that the driver can use to make auditory comments regarding incidents or other important issues (e.g. concerning the navigation system). This is a unique feature of the DAS, compared to other DAS used in euroFOT. The trigger box has the following functions:

- Record audio comment
  After audio comment button is pushed, an audio comment is recorded. The default time setting is 15 seconds and can be restarted every time the measurement system is in state "running". The start and end of an audio comment is announced by a double beep.

- Record driver familiarity with the route Button Route status
  Initially, the route status is set to 0. After start up of the measurement system, the system beeps and blinks to advice the driver to select whether the route is 1) known or 2) unknown.

Figure 23: Placement of DAS in a Daimler vehicle.
This figure is adapted from euroFOT-Daimler installation guide.

2.2.4.4 Test and quality check performed

BMW and Daimler follow their standard internal their company tests. The details of the tests are not described here but can be found in Deliverable D2.2 (Saint Pierre, Kessler and Malta 2011).

To monitor that all FOT-vehicles and data acquisition systems are working properly at any time, a web-based monitoring page was also built by MBTech for Daimler (Figure 24). No such tool is used by BMW.

Figure 24: Web-based monitoring tool at Daimler.
This figure is adapted from D5.2.

2.2.5   Italian VMC

Unlike the other VMCs, the Italian VMC does not log any CAN-data and therefore does not use any sophisticated data acquisition system. The data collection in the Italian VMC relies only on questionnaires. The drivers provided their feedback by answering specific questionnaires or filling in forms related to specific events that happened during the FOT. The subjective data collection is outside the scope of this deliverable. However, how the questionnaires data is stored and managed will be discussed briefly in the next chapter.

3 Database and Data Storage

This chapter describes the work of WP3400 towards achieving objectives 2 and 4 of SP3 (see Chapter 1). Section 3.1 explains the working method of WP3400, and Section 3.2 describes the database and storage solutions used in euroFOT.

All kind of data collected during the FOT experiment that are relevant for analysis, are stored in the database or data storage server at each VMC. This includes data from in-vehicle data acquisition systems, but also data from other sources, for instance participant-related data and subjective data.

3.1 Method/Procedure Used

The common process used to select and finally implemented the database and data storage solution in euroFOT is shown in Figure 25, from the first step gathering all requirements until implementation and test.
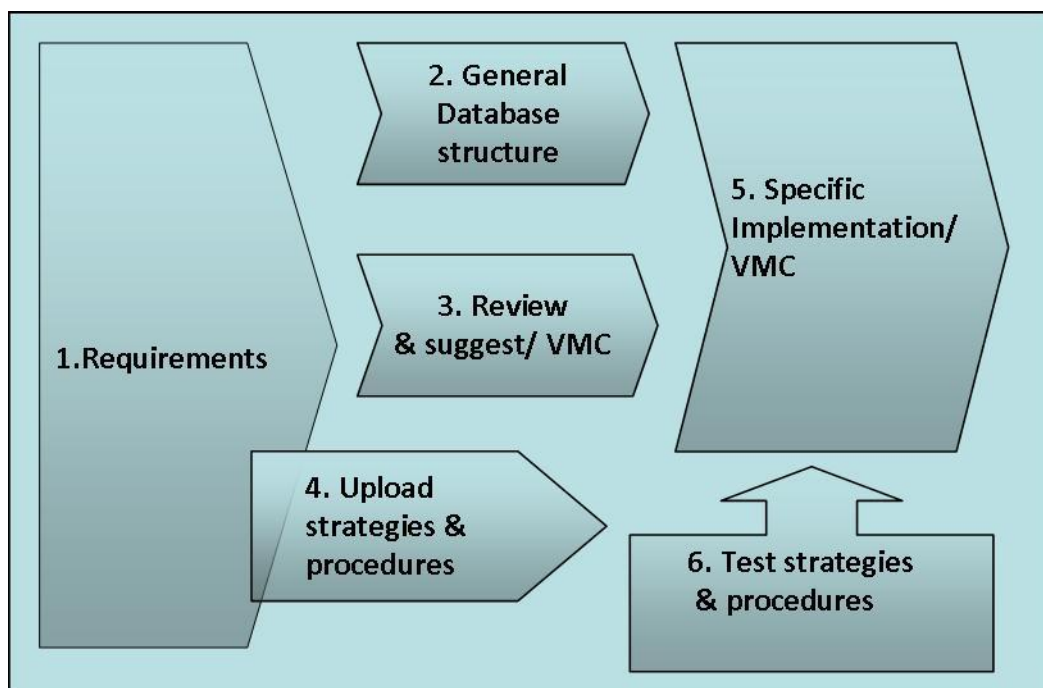
Figure 25: The common process for database and data storage.

The requirements (Task 1 in Figure 25) were gathered from the different VMCs in the beginning of the project, using common templates, with the aim to define, in a unified way, the data storage and processing chains used in different VMCs. The requirements specification supports the description of agreed, common definitions and requirements, and also allows the description of VMC specific requirements.

An attempt to make a general database structure (Task 2 in Figure 25) was made, but this proved to be not efficient to be implemented in the different VMCs. Therefore, each VMC has had to make modifications to the structure to suits their specific need.

Similarly (for Task 3 in Figure 25), when choosing database engine, fileserver, etc, specific VMC needs and constraints have to be taken into account and thus suitable solution(s) were chosen at VMC level (as these decisions, to certain degree, also depend to the IT infrastructure of the different partner organisations).

Tasks 4-6 of Figure 25 are summarised in the next section. Guidelines for upload strategies and procedures as well as test strategies and procedures are discussed and made together. However, considering the collected data types, data format, as well as resources, the implementation was done separately. Within SP3, it was decided that:

- IKA, CEESAR, and ICCS work together to implement the data management framework to be used in German-1 OC and French VMC. Note: German-1 OC and French VMC use the same CTAG datalogger II.
- Chalmers, Volvo and VCC work together to implement the solution for Swedish VMC.

Outside SP3, the partners in German-2 OC work together to implement the solution for German-2 OC and the same applies in the Italian VMC.

3.2 Database and Storage Solutions Used in euroFOT

The table below is summarizing the different implementations in the VMCs.

Table 11: The database and data storage solutions used in the different VMCs.

Category	German-1 OC	French VMC	German-2 OC	Swedish VMC	Italian VMC
Estimated size of data storage	Up to 12 TB	Up to 8 TB	(rough) 25 TB	Up to 100 TB	450MB
Database engine	MySQL (for pilot) Oracle(for production)	MySQL	MS Access	Oracle	MySQL/PHP
Other storage	Mat-files in filesystem	Mat-files and video in filesystem	Mat-files in filesystem	Mat-files and video in filesystem	
Video (Y/N)	N	Y	Y	Y	N
Logger (Y/N)	Y	Y	Y	Y	N
Questionnaires (Y/N)	Y	Y	Y	Y	Y
Enrichment tools	Y	Y	Y	Y	Y
Fleet, number of vehicles	200	40	30	130	570
Estimation of number of recorded driving hours	120000	20000	16600	80000	NA

3.2.1   German-1 OC & French VMC

3.2.1.1 Data storage solution

The solutions chosen for the German-1 OC & French VMC are similar (in terms of software architecture, processes and database schema), but is different in certain details. The similarity is mainly driven by the use of the same data logger (CTAG logger for CAN-only logger). At both places, it was investigated, which methodologies could be applied for the data retrieval processes as well as for monitoring and data analysis processes of the FOT. Based upon duration, number of vehicles and vehicle distribution, German-1 OC and French VMC classified their FOT scenarios (Figure 26) and selected a mostly automated processing chain for all three project phases (data retrieval, pre- and post-processing including enrichment, result evaluation). The automated approach in all steps of the project made specialized control structures and well traceable process interactions essential.
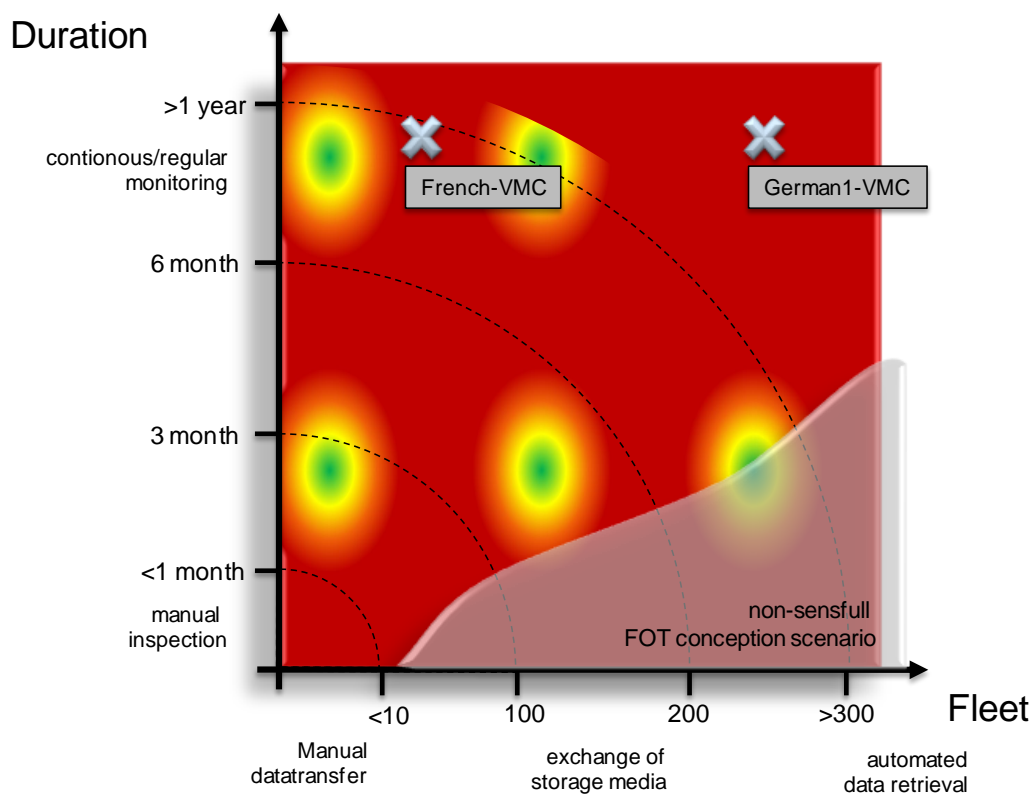
Figure 26: Classification of French VMC and German-1 OC FOT scenario.

Whilst in the German-1 OC CAN only data is recorded and later enriched by geo-map data, the French VMC also is managing in addition to those data types video data. Both VMCs have started an estimation of the amount of probable incoming data, based upon the data recording requirements of SP6 as far as these were available. In addition to that, signals being of probable interest for SP6 were added to the amount of recorded data as well. The initial data amount estimations lead to the need of investigating probable optimization potentials. Therefore optimization on process for the incoming data amount from the field in the very first beginning of the project as well as optimization for processing (pre- and post-processing) and storage structuring had to be identified.

The results of the data amount estimations have been reviewed and were verified to decrease uncertainties during experiments prior to the start of the pilot of the FOT.
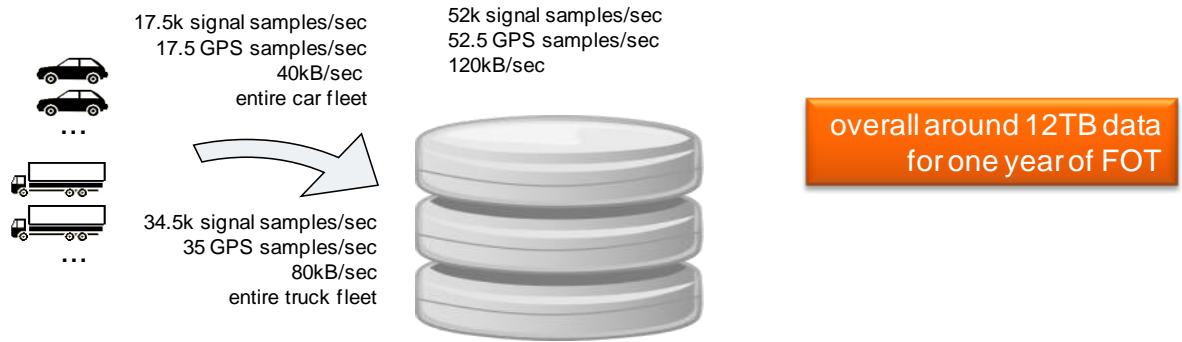
Figure 27: Data amount estimation for the German-1 OC.

The data estimation for the German-1 OC only refers to CAN recorded data and position information (GPS). It is differentiated between trucks and cars as the usage and activity levels for both vessel types are expected to be different. For both vessel types around 60 to 100 signals are being recorded. Depending on the volatility of the recorded signal, different sampling frequencies were chosen to reduce the amount of data being recorded. Trucks are estimated to run approximately 10hrs/working day, while cars will operate around 3hrs/day. Figure 27 shows the results of the overall estimation, normalized to a continuous data stream of 24/7.

Both VMCs have decided, to reduce the amount of stored data by supporting different sampling frequencies for the signals in the field and on the storage system and to avoid explicit oversampling. This approach offered the opportunity to reduce the amount of data storage needed to around 20% compared to a solution, where all data is stored with the highest sampling frequency from field.
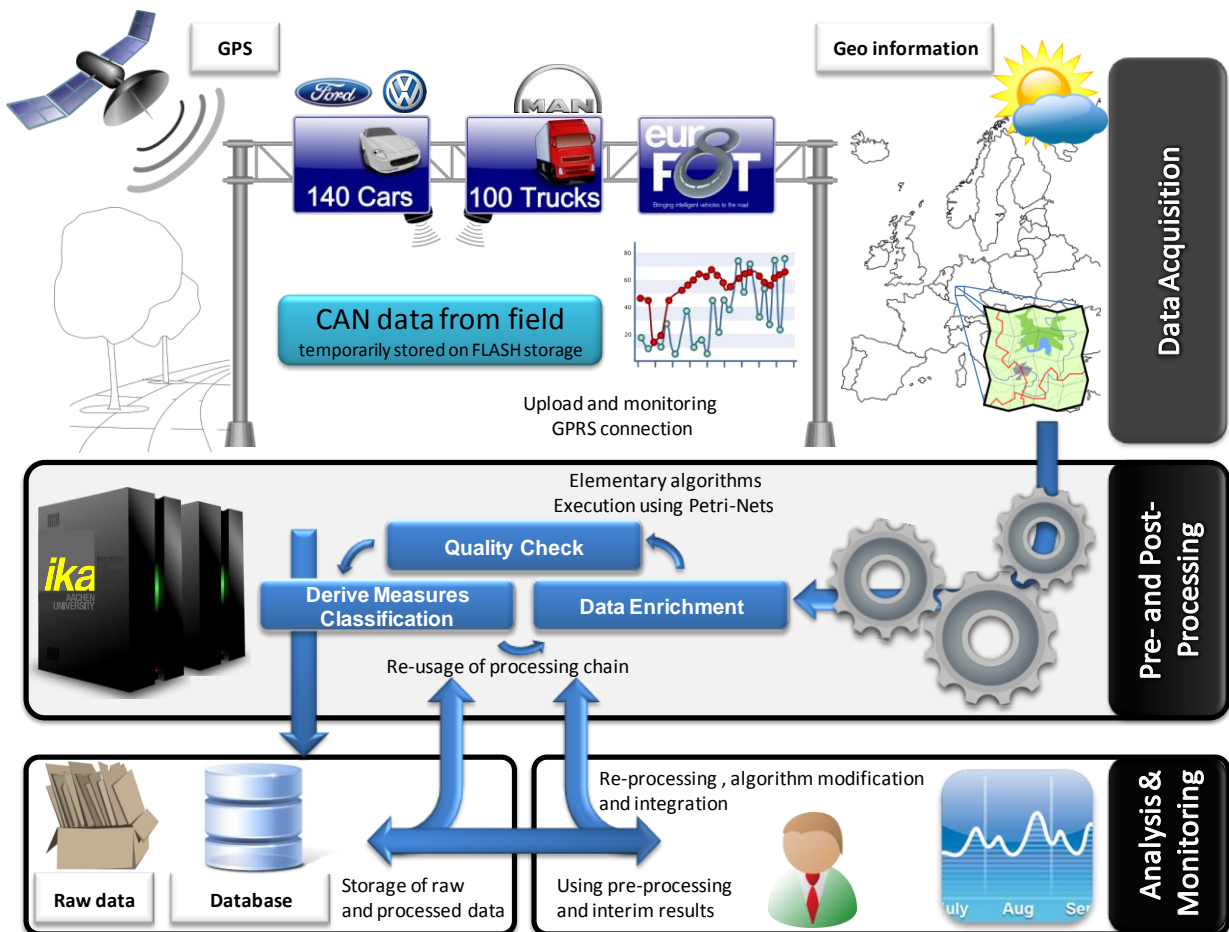
Figure 28: Framework.

To prevent algorithms to become more complex by respecting different sample times, a framework (Figure 28) - encapsulating the algorithm execution - ensures homogenous time samples during pre- and post-processing. The operation of the framework is completely based on a version-controlled configuration system: all data artefacts, their definition, relationships and user operations or algorithms which create or alter them, are stored in what is called "Trip Data Model".

Data and is stored in two different ways: file storage and relational database (Table 12). Depending on type of data and configuration, the same information can be stored either in a file, in a relational database, or in both. Backups are also implemented at both VMCs, and for both types of storage.

Table 12: Data storage at German-1 OC and French VMC.

	Meta Data (TripDataModel)	Video (only French VMC)	Raw Data	MATLAB Raw	Pre-processed	Processed
File Storage	Version-controlled structures in files.	Binary files, h.264, as acquired.	Raw binary acquisition files (CAN & GPS)	Same information as Raw data, converted in Framework's Trip Structure (MATLAB Format)	Cleaned-up, Synchronized & Harmonized data in Framework's Trip Structure (MATLAB Format)	Pre-processed + all derived and manually coded artefacts (derived measures, events, situations).
Database Storage	Meta data in relational model	N/A	N/A	N/A	N/A	Processed data in relational model, suitable for querying and analysis. Signals storage in DB is optional.
Backup	As files	As files	As files			As DB

In file storage, Trip-related data (signals and derived artefacts) is stored as MATLAB files, following a structure which is relevant for both Raw data and Pre-processed or Processed data. The Framework provides all sort of tools to manipulate or display this data.

The framework also ensures permanent consistency between file storage, database storage and compliance with Trip Data Model. To do so, it integrates a complete database management tool, which creates and maintains automatically the database, using the Trip Data Model.

This flexible approach even allows the framework to work with or without database server. A feature allows the creation of a subset of pre-processed data, based on any filtering criteria, with which it is possible to efficiently prototype and test further evolution of the Trip Data Model and calculations, before "pushing" the evolutions in the production environment.

Storage engines and server solutions chosen vary: the French VMC has decided to work with dedicated servers and MySQL (MySQL:: The world's most popular open source database, 2011) for the database, whereas the German-1 OC uses virtual machines servers and an Oracle database system (Oracle | Hardware and Software, Engineered to Work Together, 2011).

Figure 29: Database schema at German-1 OC and French VMC.
This schema is split into two to make it readable (see the next two figures).

Figure 30: The left part of the database schema at German-1 OC and French VMC.

Signals Layer

SignalsTables ▼
- 🔑 id INT
- ◆ name VARCHAR(45)
- ◆ frequency FLOAT
- ◆ description VARCHAR(512)
- **Indexes** ▶

MetaSignalInfo ▼
- 🔑 id INT
- ◆ signalname VARCHAR(45)
- ◆ tablename VARCHAR(45)
- ◆ columnname VARCHAR(45)
- ◆ unit VARCHAR(150)
- ◆ datatype VARCHAR(45)
- ◆ samplingfrequency FLOAT
- ◆ originalfrequency FLOAT
- ◆ description VARCHAR(512)
- ◆ resolution VARCHAR(45)
- ◇ defaultvalue DOUBLE
- ◇ minvalue DOUBLE
- ◇ maxvalue DOUBLE
- **Indexes** ▶

MultipleInstances

name_in_SignalsTables ▼
- 🔑 tripID INT
- 🔑 time FLOAT
- ◇ NOcolumnsDefinedInTDM ENUM(...)
- **Indexes** ▶

1..*

…otation ▼
- …VARCHAR(512)
- ▶

Process Layer

Processes ▼
- 🔑 id INT
- ◆ name VARCHAR(45)
- ◆ scriptFile VARCHAR(64)
- ◆ author VARCHAR(45)
- ◆ description VARCHAR(128)
- ◆ comment VARCHAR(512)
- **Indexes** ▶

ProcessInOutParams ▼
- 🔑 id INT
- ◆ procID INT
- ◆ name VARCHAR(45)
- ◆ type VARCHAR(45)
- ◆ variable VARCHAR(45)
- ◆ composition BOOLEAN
- ◇ value DOUBLE
- **Indexes** ▶

1   1..*

VideoFiles ▼
- 🔑 tripID INT
- ◇ videoFilesFolder VARCHAR(25)
- **Indexes** ▶

1   1

Situation Layer

MultipleInstances   1..*

name_in_SituationsTables ▼
- 🔑 id INT
- 🔑 tripID INT
- ◆ type VARCHAR(45)
- ◆ starttime DOUBLE
- ◆ endtime DOUBLE
- ◆ duration DOUBLE
- ◆ length DOUBLE
- ◆ comment VARCHAR(512)
- ◆ columnname_1 (SituationsCalculationsAttrs) DOUBLE
- ◆ columnname_2 (SituationsCalculationsAttrs) DOUBLE
- ◆ columnname_N (SituationsCalculationsAttrs) DOUBLE
- **Indexes** ▶

SituationsCalculations ▼
- 🔑 id INT(11)
- ◆ name VARCHAR(45)
- ◆ tablenameid INT
- ◆ scriptfile VARCHAR(128)
- ◆ occurencesvariable VARCHAR(128)
- ◆ description VARCHAR(512)
- ◆ comment VARCHAR(512)
- **Indexes** ▶

SituationsCalculationsAttrs ▼
- 🔑 id INT
- ◆ situationcalculationid INT(11)
- ◆ name VARCHAR(45)
- ◆ columnname VARCHAR(45)
- ◆ type VARCHAR(45)
- ◆ variable VARCHAR(85)
- ◆ unit VARCHAR(150)
- ◇ display BOOLEAN
- ◆ description VARCHAR(512)
- **Indexes** ▶

SituationTypeAttrs ▼
- 🔑 id INT
- ◆ situationtypeid INT
- ◆ situationcalculationid INT(11)
- **Indexes** ▶

SituationsCalculationsInputs ▼
- 🔑 id INT
- ◆ situationcalculationid INT(11)
- ◆ name VARCHAR(45)
- ◆ type VARCHAR(45)
- ◆ variable VARCHAR(85)
- ◇ composition BOOLEAN
- **Indexes** ▶

SituationTypeVariables ▼
- 🔑 id INT
- ◆ situationtypeid INT
- ◆ variable VARCHAR(45)
- ◆ modality VARCHAR(185)
- **Indexes** ▶

1..*

SituationTypes ▼
- 🔑 id INT
- ◆ name VARCHAR(45)
- **Indexes** ▶

SituationsTables ▼
- 🔑 id INT
- ◆ name VARCHAR(45)
- ◆ description VARCHAR(512)
- **Indexes** ▶

Figure 31: The right part of the database schema at German-1 OC and French VMC.

The following scheme presents the used data types and the different processing steps after data has been uploaded on the server. As a first step the uploaded data is converted to a MATLAB file, in order to start the following processing steps. Afterwards these data is enriched with data from the digital map and the needed events, PIs for data analysis. These data is stored on a file server per trip-basis. As soon as data processing is accomplished the data is uploaded on the database. For this the processed MATLAB files are translated by means of the database schema and stored and stored on the SQL database.

The data stored in the database has different sampling rates, in order to reduce the amount of data to be stored on the database. The following sampling rates have been defined (based on requirement for data analysis):

- 1Hz: Status information (turn indicator, system status etc.), Map data(road type, speed limit etc)
- 10Hz: vehicle dynamics (speed, longitudinal acceleration etc.)
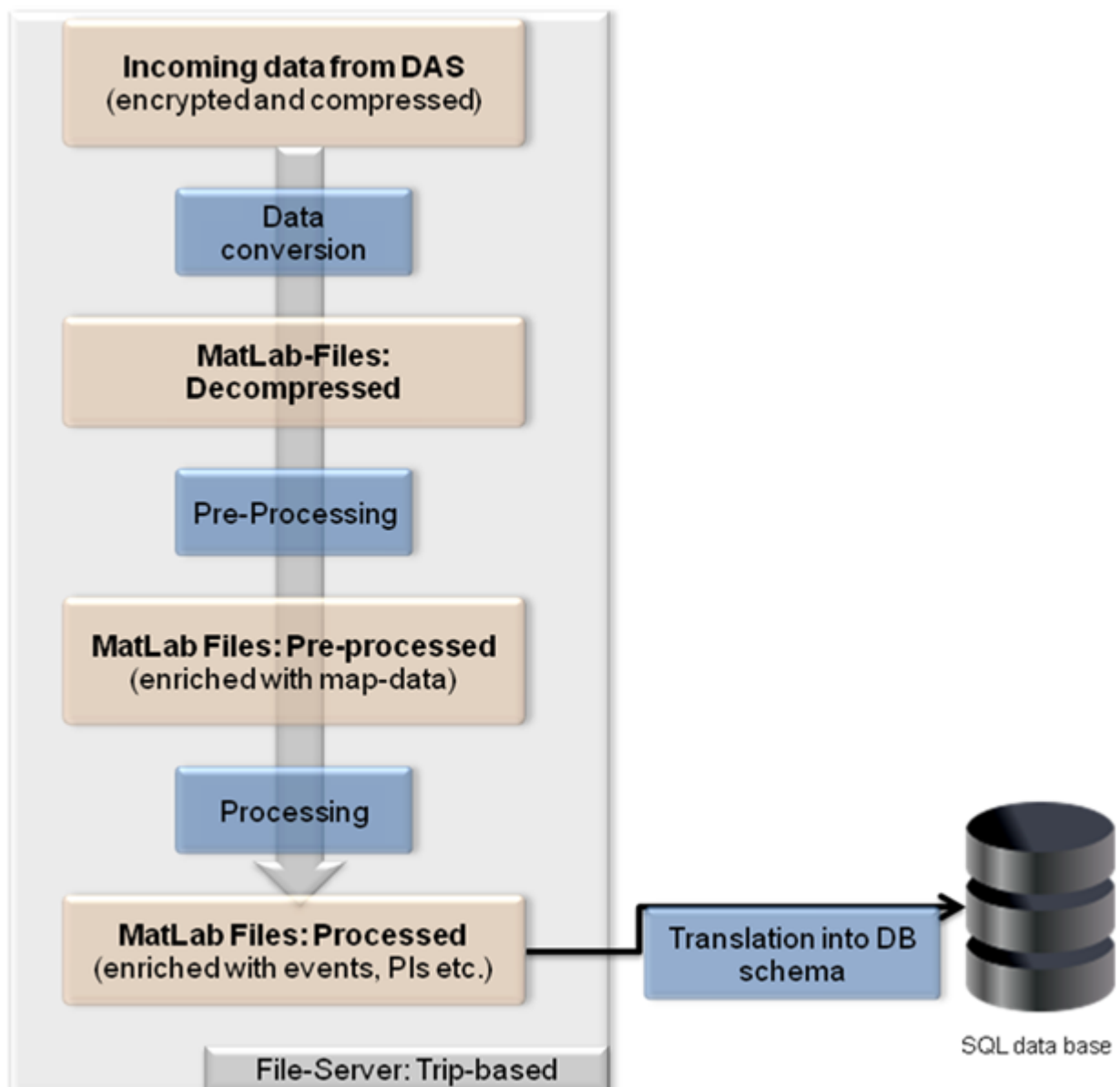
Figure 32: General layout of data processing at German-1 OC and French VMC.

Deliverable D3.3:
Data management in euroFOT
<Version 1.0>
55

3.2.1.2 Upload and enrichment tools

To reach the aim of a mainly automated processing chain, the implemented approach of the German-1 OC and French VMC distinguishes strictly between the framework and embedded algorithms in an algorithm pool, encapsulated by the priory mentioned framework. The framework was designed to be of universal use for the data enrichment procedures and algorithms execution in all steps of the project (pre-, post-processing and data enrichment) follow the same structure and are described by the Trip-Data-Model (TDM).

The TDM describes all artefacts handled by the implemented framework. Artefacts of the meta description are signals, data structures, attributes, events, situations and algorithms (including the enrichment algorithms) on meta level. This meta level information is used by the framework to bind incoming data and external data sources (like geo map data) together with the algorithms from the algorithm pool.

It determines a correct execution order of the algorithms and keeps track of evolving algorithms. For this purpose, versioning information is included in all generated output artefacts of the framework, which guarantees consistent data at every time of the project. The versioning information is as well used to determine, if already processed data sets require reprocessing due to evolved algorithms. Hence, when evolutions are made either inside an algorithm, or in the data model itself, these evolutions are automatically detected, and if validated, result in automated database structure evolution, and refreshing of data, where only impacted steps are re-evaluated.

Beneath the data analysis tool used in both VMCs, all enrichments are executed by embedding algorithms to the TDM.

Figure 33: Analysis tool.

Figure 34: Pre-processing.

3.2.1.3 Lessons learned

Designing the data storage and database structure faced several challenges. Initial strategy and specifications related to database configuration were updated and amended many times during project evolvement. New challenges emerged which we did not think about at first place and needed to overcome. So the experience obtained by the whole activity on the project provided us invaluable knowledge in huge dataset manipulation techniques.

Limitations and issues
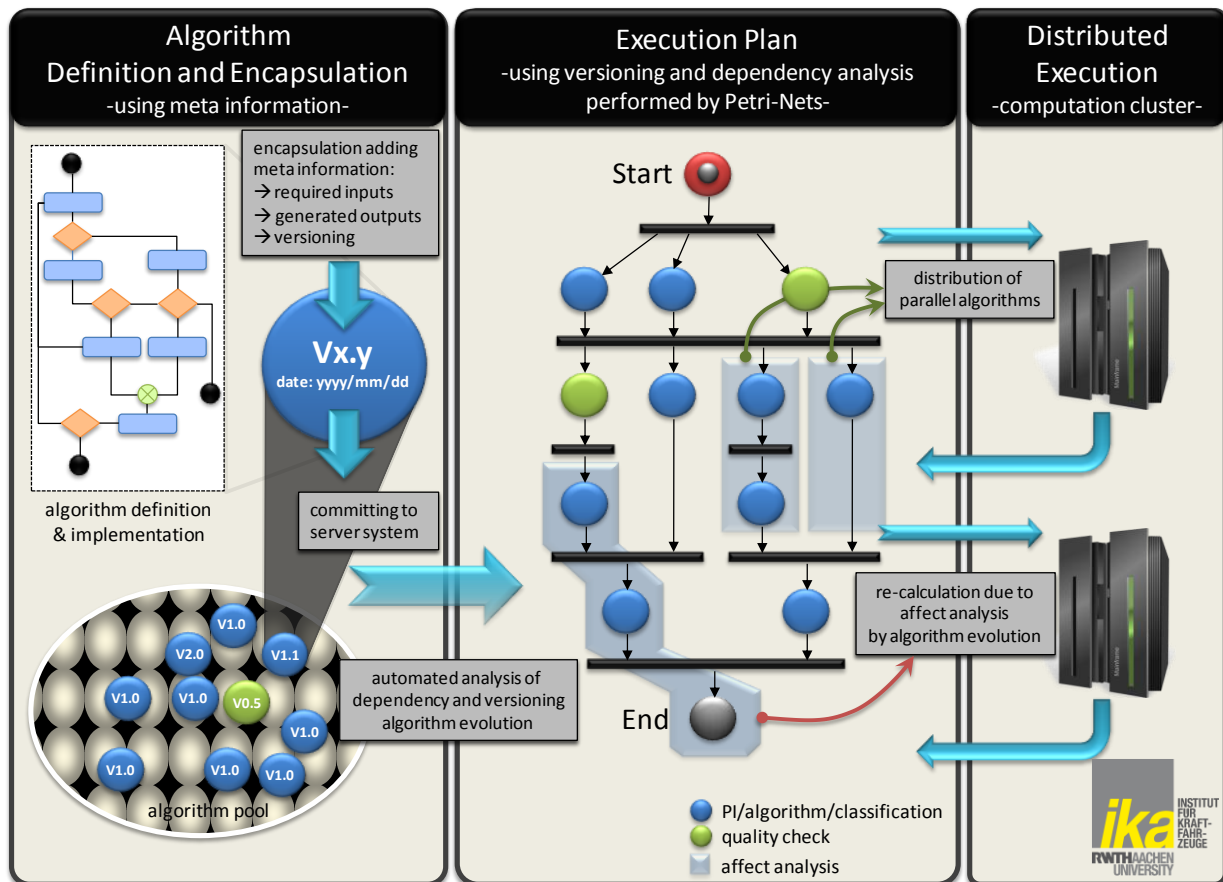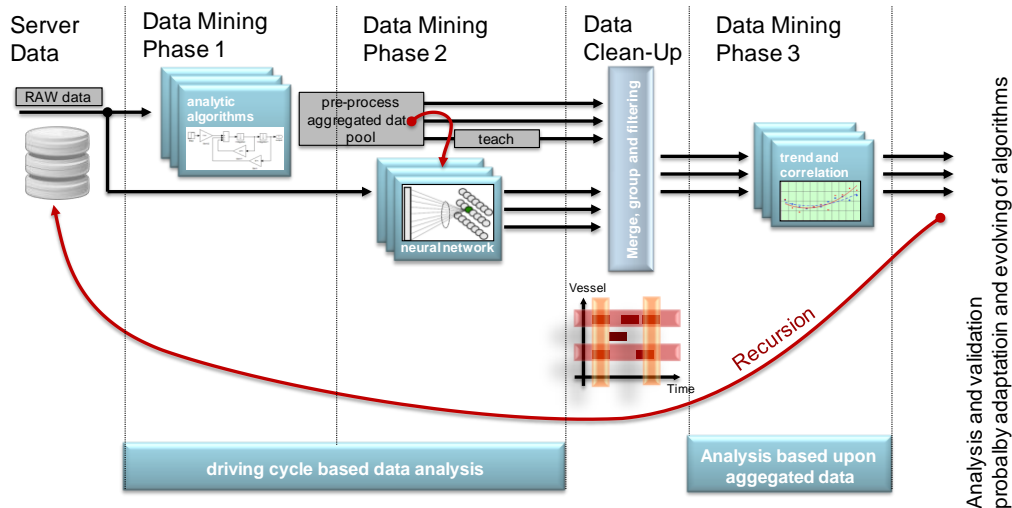
Main difficulty was the design of the database schema and the methodology on storing data in a relational database system. The time it takes to upload large datasets from field operation tests is critical. That is, if the sum up of all the steps that needs to be performed for data uploading is for example at the same speed as it was recorded, then 20 hours of driving data would take up to 20 hours to upload. This is unacceptable and not reasonable since data collected from multiple drivers may need to be uploaded in one day. Also the way that analysts need the data influence the design of the database schema. It is necessary to avoid performing joints between tables with millions of rows of data during the analysis phase to have them presented in useful way. Another issue was the foreign keys which needed for database integrity. So creating an optimal database schema is optimal for database's desirable operation. Additionally it would be really difficult and time consuming to gather all data under a certain hard disk in order to upload them to the database. Finally we should be able to prevent a bad usage of the API (functions that implement the interaction with the database) which would create an inconsistent database. So a lot of thought and experimenting needed to create a fast, reliable, consistent and flexible relational database.

Successes

The careful design of the database schema was an urging and important work. During the process of data storage the upload is going on trip per trip which includes original signals, derived metrics and aggregated data related to each trip. All *meta* information is also stored in the database. This includes for instance the description of experimental or aggregated data, as well as the description of the complete data reduction process and its history. As a result, although original trip files and video files are kept in a file-based storage, the database is self-documented and hence contains all the information necessary for data analysis. The

interaction with the database is taking place in two phases. Meta data describes all the objects in trip data, as well as the relationships between them and trip data corresponds to the original data, as well as the data which is derived from it during data reduction. So using the meta data we create the tables/columns in such a way that data which are supposed to be huge for instance events and their attributes are stored in single tables. Thus, we prepare the data in compatible form and use database specific techniques, like sql-loader, to perform fast and reliable uploading. Also the above separation gives us the ability to create different types of users which will have different rights to database manipulation. The structural characteristics of the database are defined by SQL statements that comprise the Data Definition Language (DDL- create/alter/drop).   If the above statements fail during the execution we cannot rollback the database to a previous consistent state. So we create another protection layer by giving the ability to the user to make an informative inquiry of the upcoming changes to the database during meta data process. The overhead and the speed bottleneck created by foreign keys are overcome by transferring the control from the database to our library for huge tables. The users also have the ability to remotely send/upload the data to a specified database, simplifying a lot the data upload process. Finally the structure of the database gives the analysts the ability to define filters and directly access the data corresponding to the entire dataset (i.e., data from all the trips) which are relevant and necessary to test each hypothesis.

Performance

With respect to quality

Data quality on sample and signal basis is performed directly after data conversion. The checks on trip basis are performed within the pre-processing step. Based on an error-handling integrated in the framework, errors due to missing or wrong information are automatically presented, in order to identify the problem and define a suitable solution.

The way to check if data quality is acceptable or not, and the way to proceed when the quality is not enough, is different depending on where the data comes from. On the highest level there are basically two kinds of data: objective data which comes from sensors integrated in the vehicle and subjective data which comes from questionnaires filled by the drivers.

- Objective data (on-vehicle sensors)
    - CAN data
    - GPS and map data
- Subjective data (questionnaires)

The following checks are applied, in order to assess the quality of the incoming data by means of automated quality checks integrated in the framework:

- Missing data. Because of malfunctioning sensors, unfastened plugs, faulty acquisition systems or human error, there are many possible factors which cause data loss.

- Values out of range. The acquired data contains information about the driver or the environment. This information has a physical meaning, and therefore also physical bounds. When the registered values are out of bounds we may conclude that something has gone wrong. For example, if the velocity is -511 km/h it is clearly erroneous.

- Wrong dynamic behaviour of data. If the data originate from a physical signal, it has to obey the laws of physics and hence we can predict the behaviour or shape of the signal. For example, if the velocity oscillates between 10 and 100 km/h it does not behave in a physically meaningful way. Note however that a sudden deceleration

from 100 to 0 km/h indicates a hard braking event which should not be removed by some automatic attempt to remove low quality data.

- Incoherent relationships among data. Some measures in a data set are related to each other. For example, acceleration and velocity are tightly connected to each other, and when the value of the acceleration is positive the velocity should increase. Otherwise there is an incoherent relationship between those signals. In some cases, the same information is acquired with independent sensors. For example, velocity may be acquired from the CAN bus as well as from GPS. In such cases these two signals can be compared for quality assurance purposes.

<u>With respect to speed</u>

Tests were only performed on a limited amount of pilot data sets and on dummy data sets generated with the aim of testing the frameworks functionality and stability. Therefore, no information related to performance-speed can be reported here.

Future recommendations

Current database schema support data records which are trips. In the future we need to take into account the storage of several data records, from all the interacting entities, for one single event such as road network log, several vehicles DAS, mobile phone applications logs.

3.2.2   Swedish VMC

3.2.2.1 Data storage solution

Figure 35 shows an overview of the implementation, visualizing the process from data pre-processing to upload and interface to analysis.
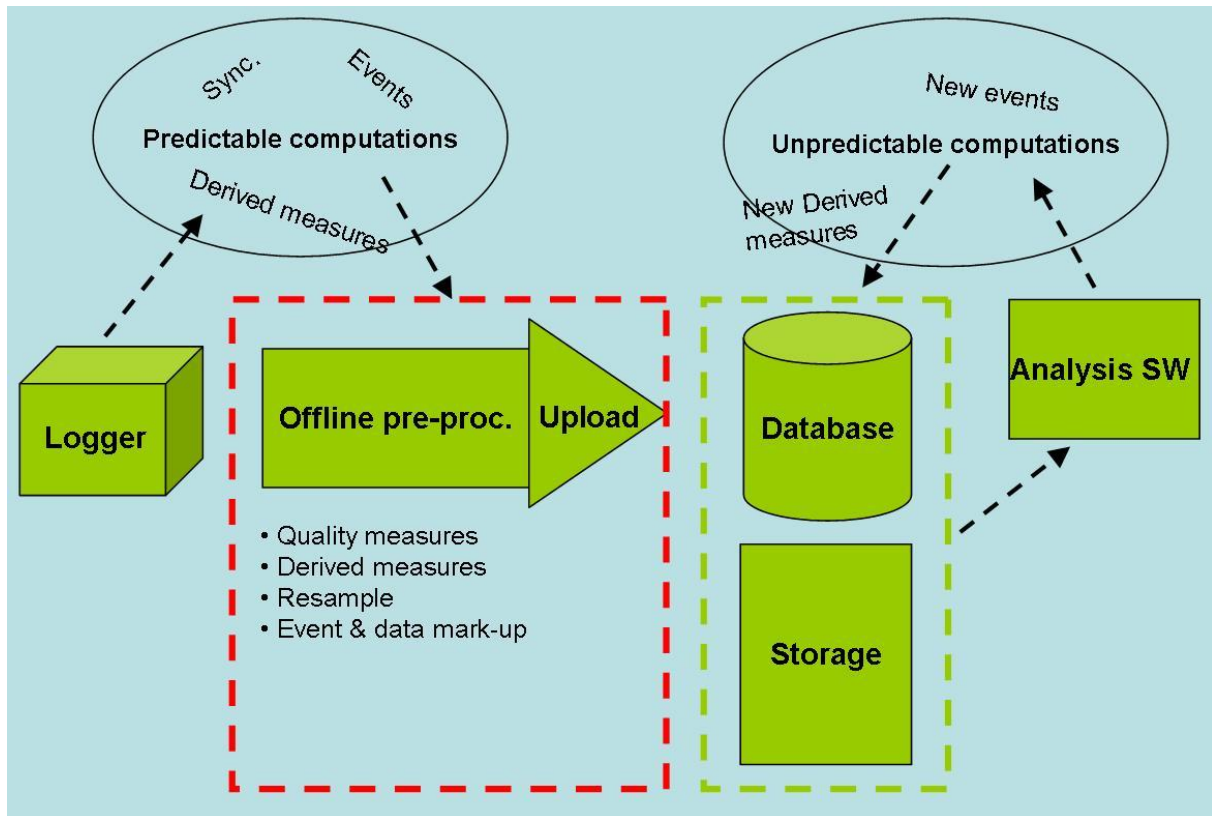
Figure 35: Overview database and data storage.

The FOT (objective) data in the Swedish VMC consists of 1) video data and 2) non-video data (CAN data and eye tracker data). Additionally, there is also questionnaires (subjective) data.

Table 13 and Table 14 show the estimate of data size from each OEM for one year data collection.

Table 13: Estimated size of time history data in the Swedish VMC

OEM	Vehicles	Hours per week	No of samples per week, 10 Hz	10 Hz in 12 months	No of measures	Bytes per value	Size for 12 months (GB)*
Volvo	30	30	32 400 000	1 684 800 000	180	4	1 130
VCC	100	10	36 000 000	1 872 000 000	180	4	1 255
Total:	130		68 400 000	3 556 800 000			2 385

*Uncompressed data

Table 14: Estimated size of video data in the Swedish VMC.

OEM	Vehicles	Hours per week	Video data size, MB per h	GB data per week	Size for 12 months (GB)*
Volvo	30	30	700	630	32 760
VCC	100	10	700	684	35 547
Total:	**130**			**1 314**	**68 307**

*Uncompressed data

Mainly based on the size estimation, the Swedish VMC has chosen the following solution for storing the FOT (objective) data:

- upload the non-video data (as per trip basis) to an Oracle database.

- upload the video data as files to a fileserver.

To connect the non-video data to video data, a link to an associated video file of a trip is stored in the Oracle database as part of each trip's record.

The rest of this section will discuss the processing and upload of objective data. The questionnaires data are planned to be stored in the Oracle database (tables have been prepared, see Figure 37) and the fileserver (saved as files like MS Excel).

The overall upload task consists of several steps. Generally, the flow of FOT data from the vehicle(s) to the Swedish VMC centre can be described as in Figure 36. From the technical implementation point of view, the overall upload task can be divided into two parts, ``pre-processing´´ and ``uploading´´. The two parts are defined as follows:

- *Pre-processing (PP)* as the step between when the data being available in an office at an OEM in the form of USB-discs (directly from the field/vehicle) until the data is in a final MATLAB (.mat) format to be uploaded at the VMC centre.

- *Uploading* as the step of reading the data from the transfer disk(s) and uploading the non-video data to the Oracle database and uploading video files and .mat-files in appropriate folders in the fileserver.

A more detailed description of each part will be described later on in this section.

In order to optimize the overall data pre-processing time, it was important to reduce the number of time data is read from hard-disk and pre-processed before finally being uploaded to the database and data storage at the VMC centre. Since there are some processes that can only be done at OEM, for example decoding and extracting CAN messages, it was considered more efficient to do as many pre-processing steps at the OEM side as possible. The implementation of the needed pre-processing scripts/module was jointly done by the Swedish VMC partners (led by Chalmers).

During the pre-processing step, the non-video data from USB hard-disk is converted to a final MATLAB (.mat) format before being uploaded to the Oracle database. These MATLAB files will also be uploaded as files to the fileserver. This is done to allow analysis stage to be done through and benefit from the strength of relational database in terms of access rights, concurrency, consistency, integrity, etc as well as the easiness of flat files operation for certain types of analysis.

The chosen database engine is Oracle 11g (Oracle | Hardware and Software, Engineered to Work Together, 2011). The database schema used in the Swedish VMC is shown in Figure 37. Overall the FOT data can be divided into several groups based on whether it is a raw data, enriched data, and also based on who can access which data to ensure consistency and security. Therefore, access rights are enforced in both the Oracle database and the fileserver.
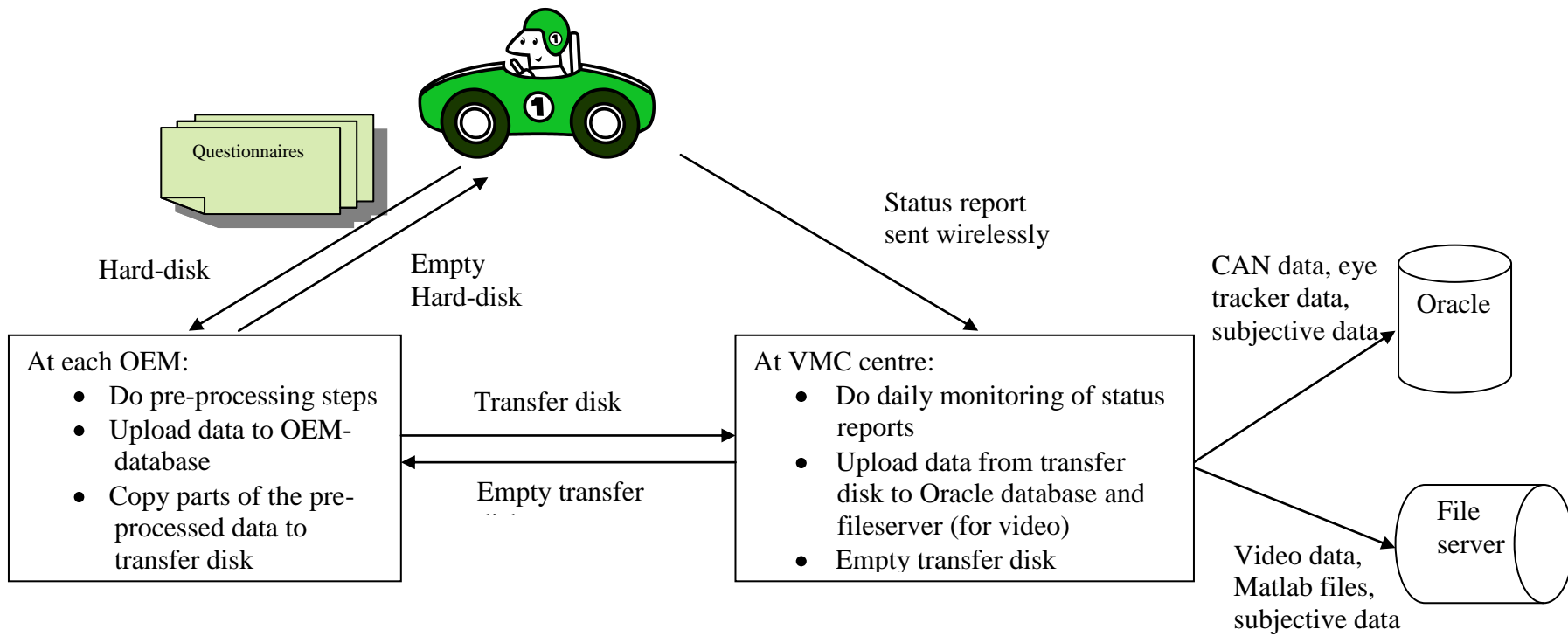
Figure 36: Overview of data flow from the vehicles to the VMC centre in the Swedish VMC.

Figure 37: The database structure used in the Swedish VMC.
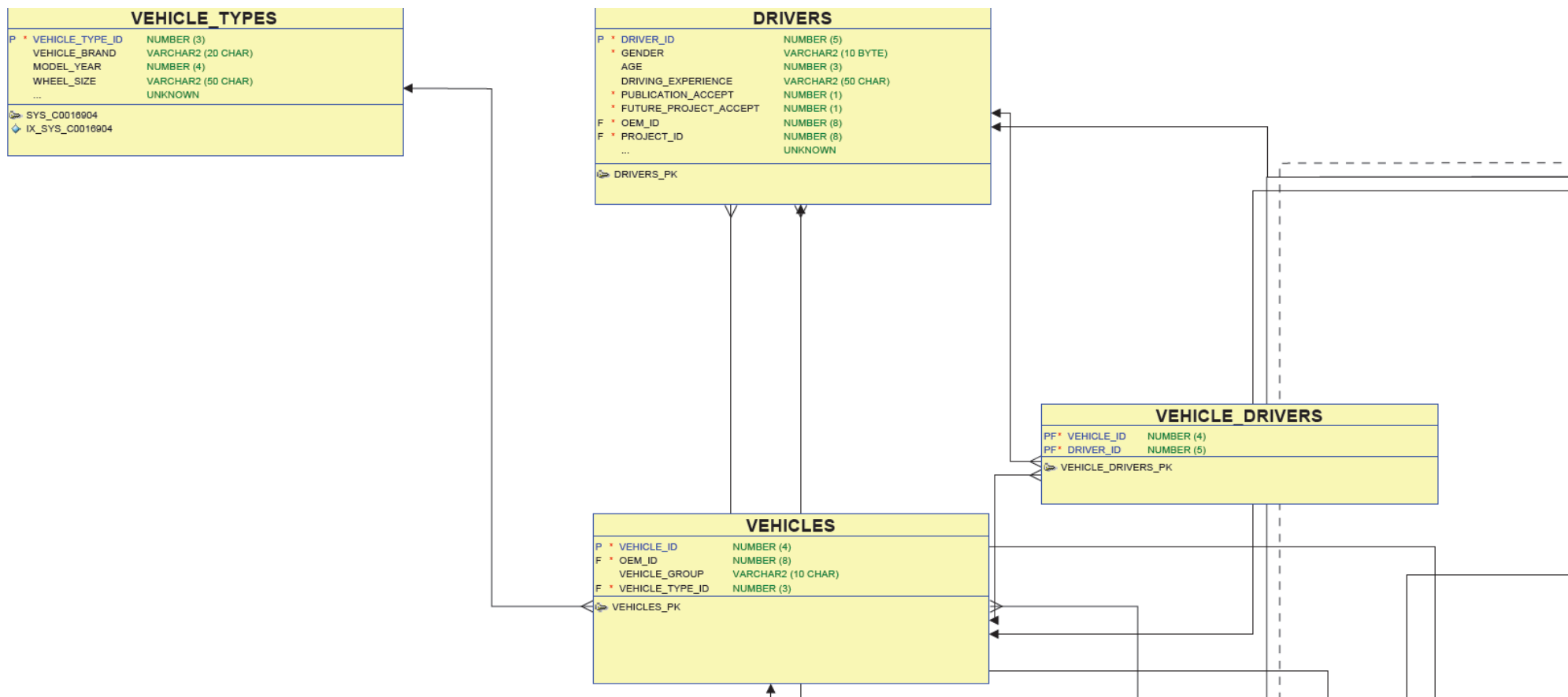This structure is split into six parts to make it readable (see the next six figures).

Figure 38: Part-1 of the database structure used in the Swedish VMC.

Figure 39: Part-2 of the database structure used in the Swedish VMC.

TRIP_QUALITY_MEASURE_VALUES

PF	* TRIP_ID	NUMBER (8)
PF	* MEASURE_ID	NUMBER (8)
P	* NAME	VARCHAR2 (50 CHAR)
	VALUE	NUMBER
	STRVALUE	VARCHAR2 (30 CHAR)
	* DETAILS	NUMBER (1)

TRIP_QUALITY_MEASURE_VALUES_PK

TRIP_VEHICLE_MEASURES

PF	* VEHICLE_ID	NUMBER (4)
PF	* MEASURE_ID	NUMBER (8)
F	TRIP_ID	NUMBER (8)

VEHICLE_MEASURES_PK

MEASURES

P	* MEASURE_ID	NUMBER (8)
	* NAME	VARCHAR2 (240 CHAR)
	DESCRIPTION	VARCHAR2 (3000 CHAR)
F	PER_SAMPLE_ID	NUMBER (8)
PF	* OEM_ID	NUMBER (8)
	...	UNKNOWN

MEASURES_PK
MEASURES_ID_UNIQUE

EVENT_MEASURES

F	EVENT_TYPE_ID	NUMBER (4)
F	REF_MEASURE_ID	NUMBER (8)
F	REF_EVENT_TYPE_ID	NUMBER (4)

EVENTS

P	* EVENT_ID	NUMBER (20)
F	* EVENT_TYPE_ID	NUMBER (4)
F	* TRIP_ID	NUMBER (8)
	* START_TIME_INDEX	NUMBER (10)
	STOP_TIME_INDEX	NUMBER (10)
	EVENT_LEVEL	NUMBER
F	* OEM_ID	NUMBER (8)
F	* PROJECT_ID	NUMBER (8)
	...	UNKNOWN

EVENTS_PK
EVENTS_IDX1

MEASURE_PARENTS

F	* PARENT_ID	NUMBER (8)
F	* MEASURE_ID	NUMBER (8)
	VALID_FROM	DATE
	VALID_TO	DATE

UPLOADED_FILES

P	* FILETABLE_ID	NUMBER (8)
	INSTANCE_ID	NUMBER (8)
	FILENAME	VARCHAR2 (50 CHAR)
F	TRIP_ID	NUMBER (8)
	COPIED	NUMBER (1)
	PROCESSED	NUMBER (1)
	UPLOADED	NUMBER (1)
	ROWSUPLOADED	NUMBER (7)
	ROWSFAILED	NUMBER (7)

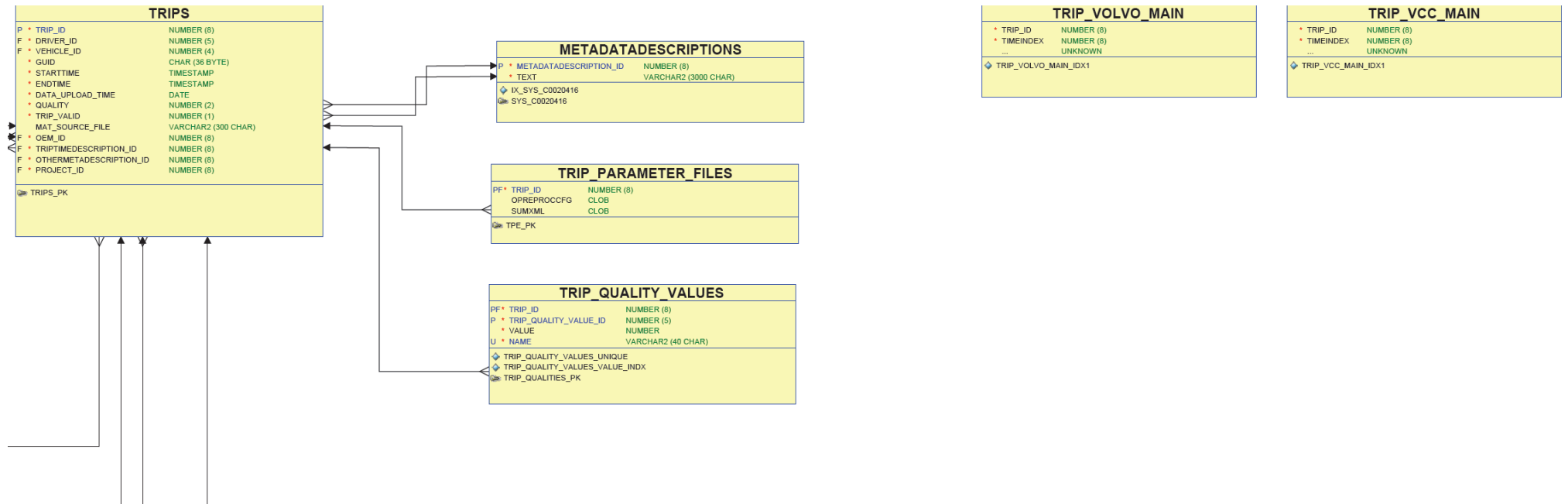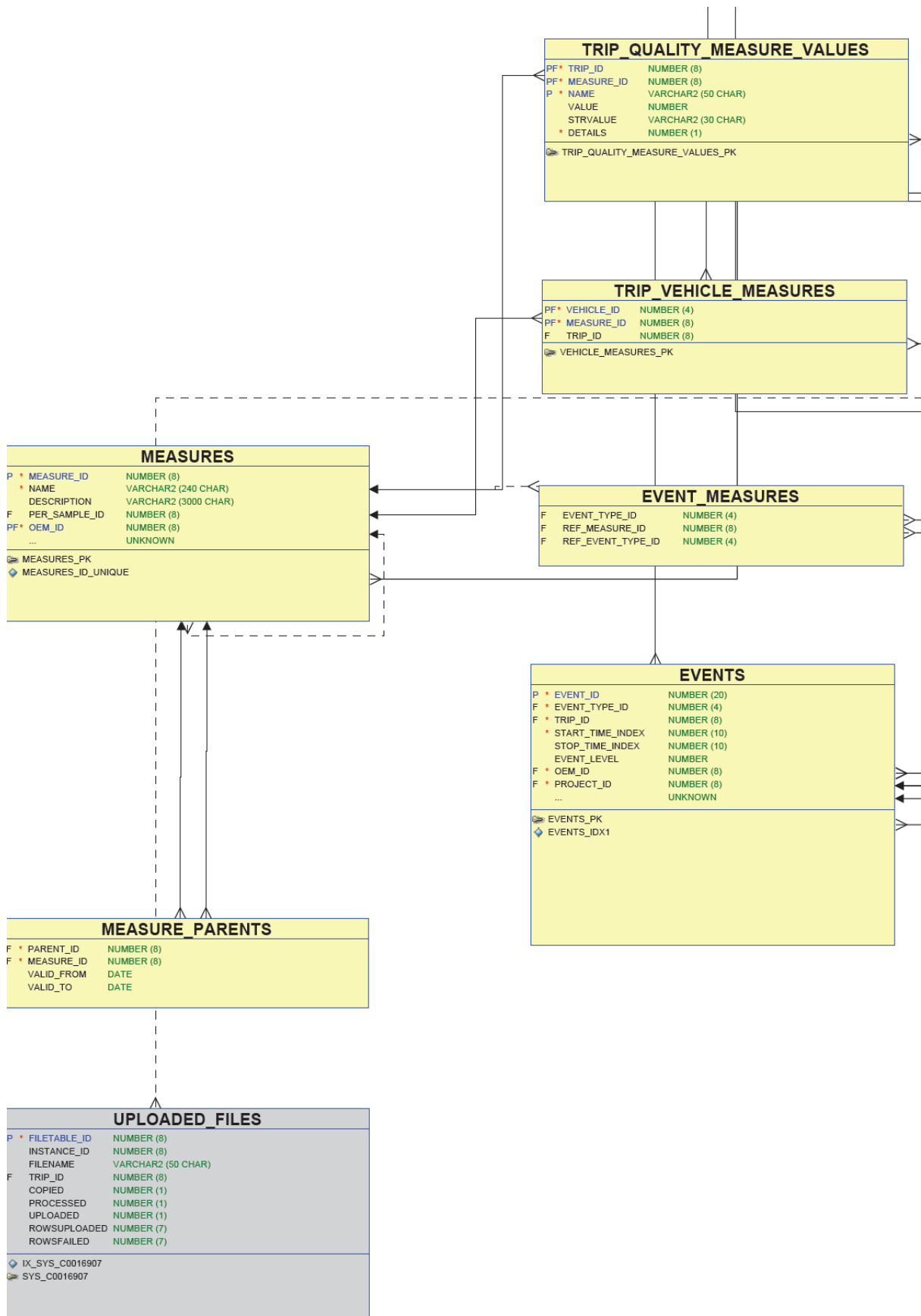IX_SYS_C0016907
SYS_C0016907

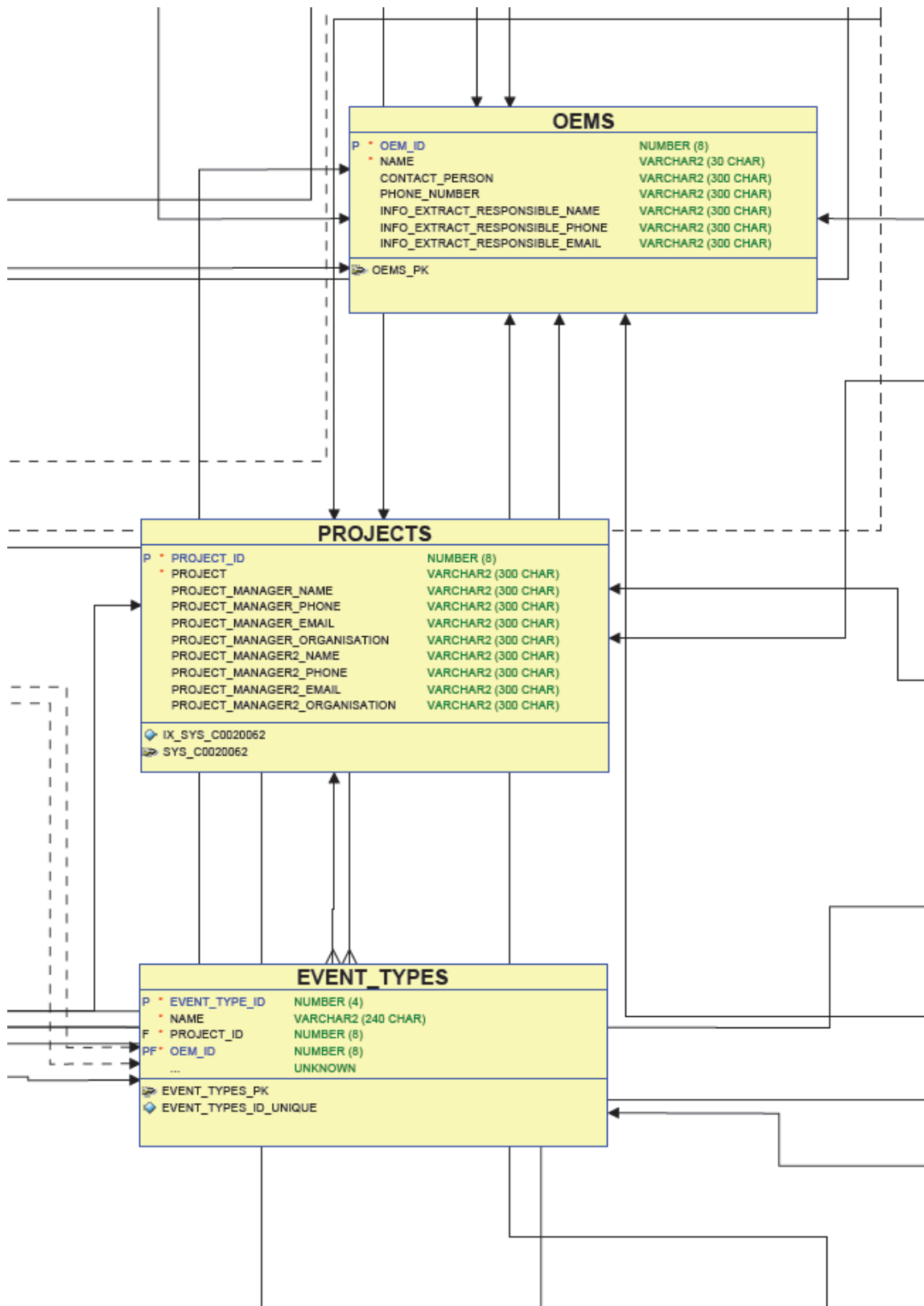Figure 40: Part-3 of the database structure used in the Swedish VMC.

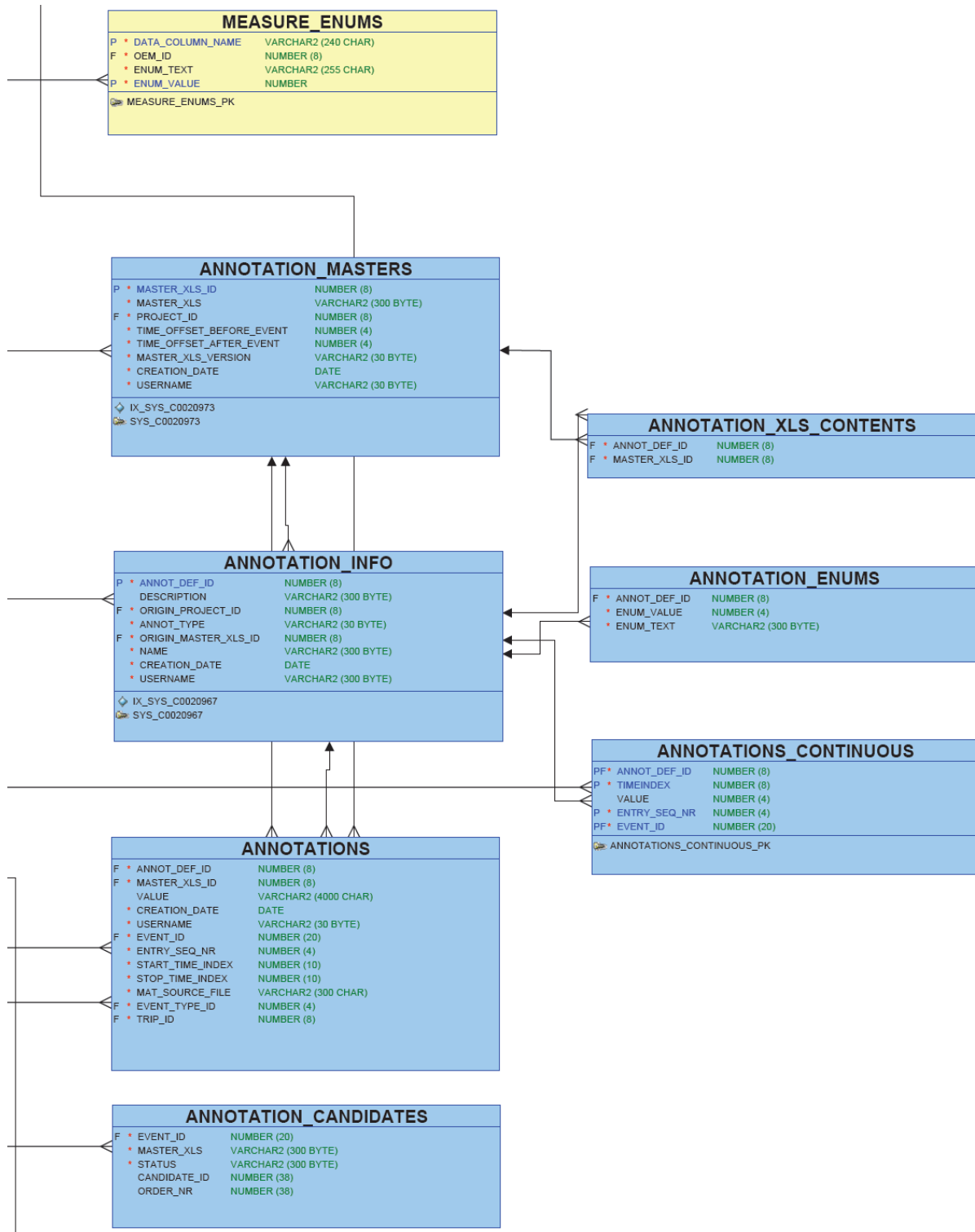Figure 41: Part-4 of the database structure used in the Swedish VMC.

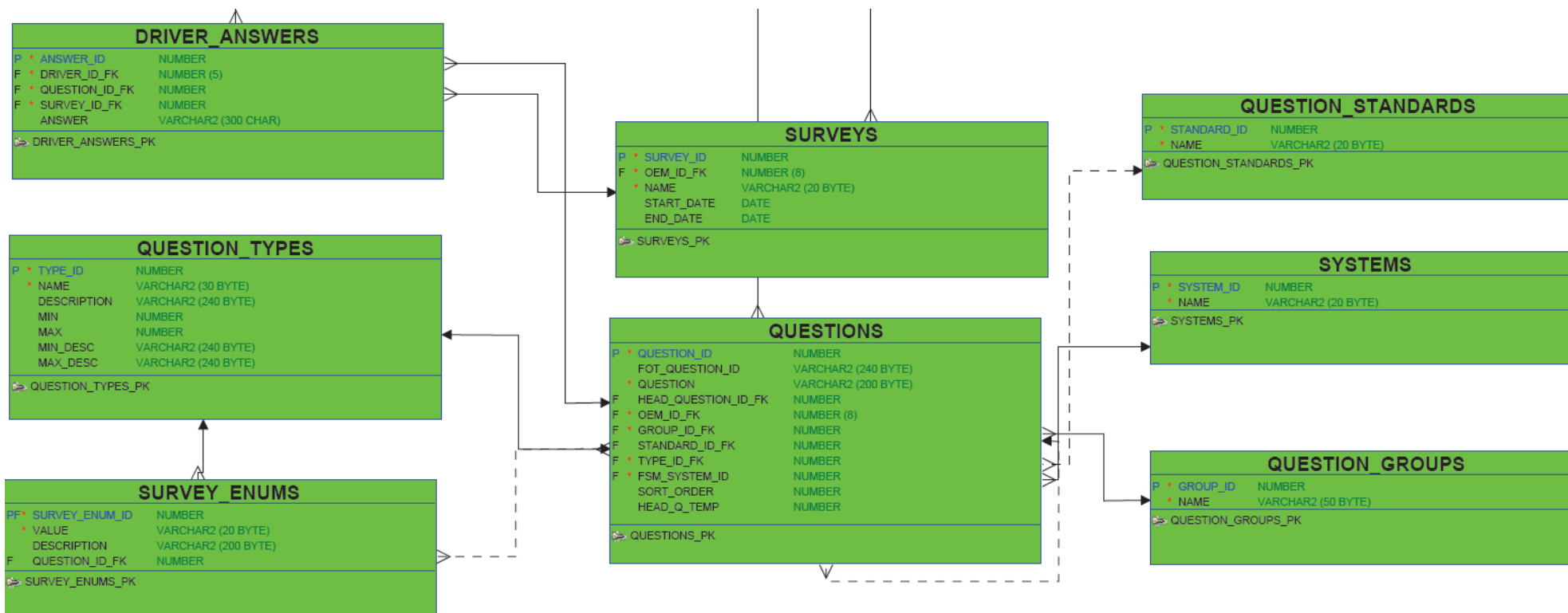Figure 42: Part-5 of the database structure used in the Swedish VMC.

Figure 43: Part-6 of the database structure used in the Swedish VMC.

The MATLAB structure that describes the data (oDBdata.mat structure)

The oDBdata is the format of data after pre-processing and before uploading to the database. Figure 44 is an overview of the structure. Note that oDBdata is the format that all functions in analysis are and will be using (this is what the queries to the database will be returned as).
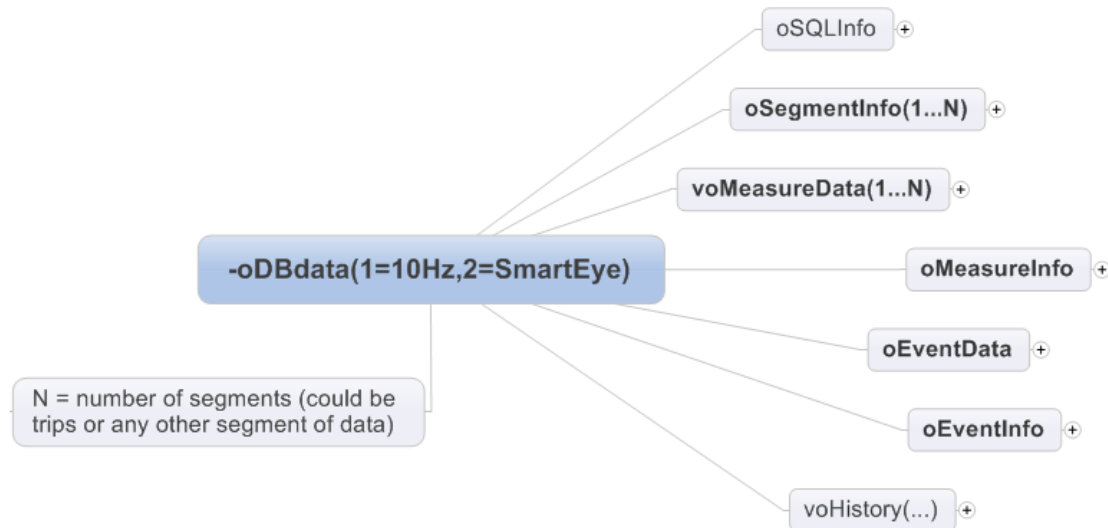

Figure 44: Overview of MATLAB struct oDBdata.mat.

oDBdata: The main level of the *struct* with two indices. One (1) for all 10Hz data and 2 for all EyeTracker data. This is the *struct* that the pre-processing produces as output and what the uploader takes as input.

oSQLInfo: Since this is also the format returned when querying the database for analysis, the information and history about how the data was retrieved and who retrieved are stored here. This is not relevant for the pre-processing and upload.

oSegmentInfo: This is called SegmentInfo since it is also used for analysis, but for pre-processing and upload it is the trip information. Here all the information about e.g. trip_ID, driver_ID, trip_start/stop, OEM-name, as well as trip quality information and references to video files, are stored. The segment info can, when returned as a query, be a *struct array*, but for pre-processing and upload this is always only one instance (corresponding to the trip).

voMeasureData: A *struct array* with one instance per segment. That is, when data is retrieved for analysis data is extracted as segments. For pre-processing the number of segments will always be one (1) and corresponds to the whole trip. This is where all measure data end up (including derived measures). There will always be a field called timeindex which is a uniformly increasing time index in 100 millisecond steps from 0 (except for EyeTracker where timeindex is the sampled time in uneven milliseconds). This is what is used for all in-trip analysis. Then one field per measure. All fields have the same length.

oMeasureInfo: Include all metadata information about the individual measures. A structure with fields of all names that are in voMeasureData. There is always a MeasureInfo for a MeasureData field. For each measure information i.e. number of significant digits, measure description, unit, enumeration description, data source, creator and creation time (for traceability), as well as per measure quality are stored. The per measure quality has three levels:

1. the overall MeasureValid (0 or 1),
2. qualityDetails (specific information that may differ between measures) and

3. PreProcQuality (a general procedure applied to all measures in the same way, but with thresholds). Also, if there is a per sample vector available for this measures, a reference to this vectors name is found here.
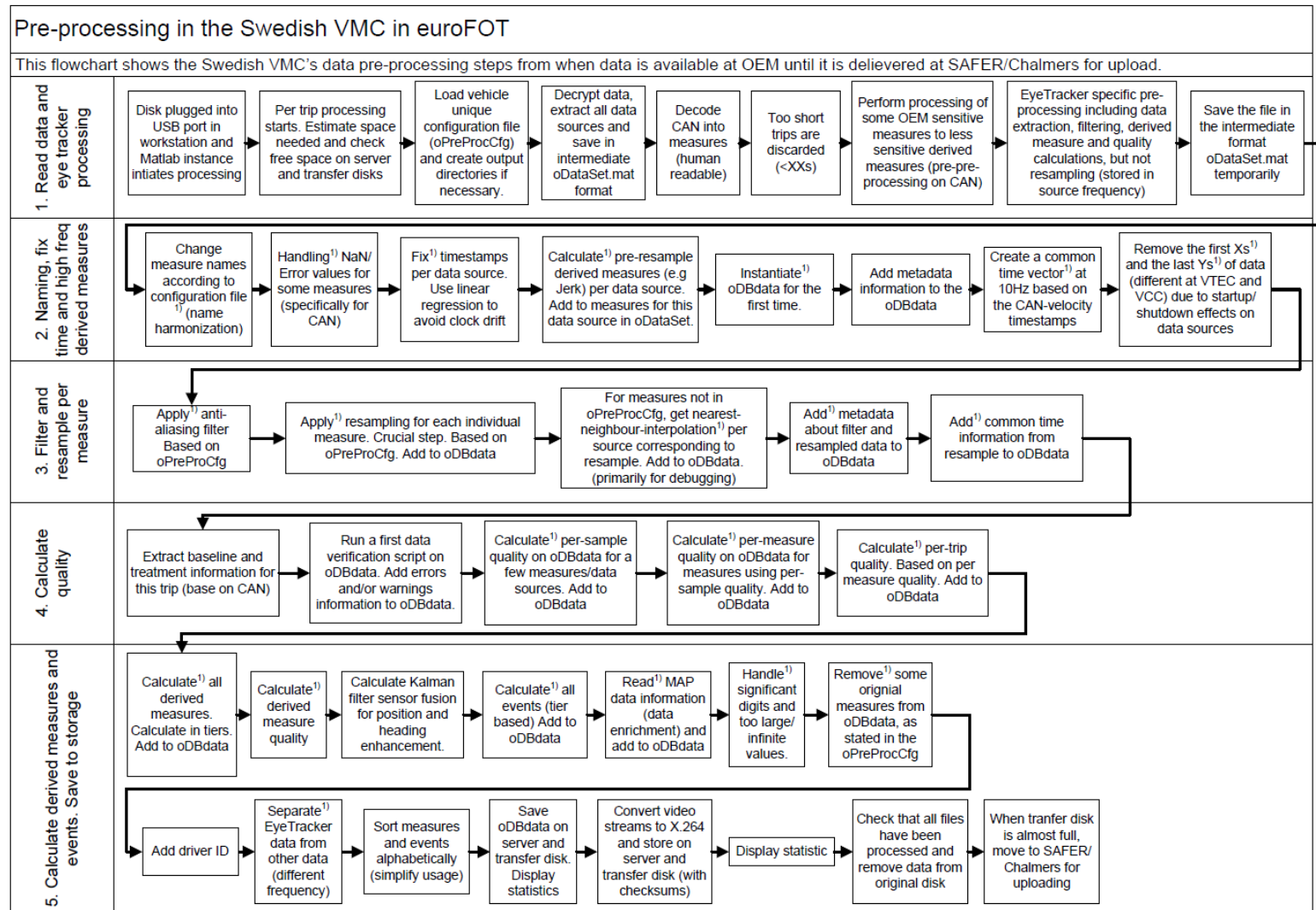
oEventData: A *struct* with one field per event-name (e.g. eCRELongAcc06). Each field is a *struct array* corresponding to each of the events found of this type for this trip. The data included is Trip_ID, StartTime, StopTime, EventLevel (value that triggered the event – specific definition for each event), Text (specific text can be added that describes each particular event – not used much), the EventTypeID (unique database ID related to the name) and the EventID (unique DB ID for this particular event). Note that the EventID is unique also between VCC and Volvo with events created at VCC with a 1 as last digit, 2 for Volvo and 3 for Chalmers.

oEventInfo: A *struct* with one field per event-name in oEventData. The following information is stored: Description, the source (script name that created, creator and creation time), the EventTypeID and information if it is a single point event or if it has duration (start/stop).

voHistory: Not used for pre-processing; only for history information about what scripts has been applied (and who applied them) when a dataset is extracted though a query for analysis.

The ``Pre-processing´´ Step

The pre-processing (see Figure 45) starts with the definition of a MEPS document for each OEM. MEPS is short for Measures Events Performance indicators and Situational variables. The MEPS is a Microsoft Excel document that includes all configuration items needed for the pre-processing. The MEPS document is then used to dynamically create vehicle specific oPreProCfg.m files by MATLAB parsing the MEPS document. oPreProcCfg is a configuration file unique for each individual vehicle and is the core in the pre-processing – everything defined in MEPS is individualized in the oPreProcCfg.mat and then used to define pre-processing for each individual vehicle. The flow chart shown here explains the main flow of the pre-processing.

Figure 45: The pre-processing steps in the Swedish VMC.

3.2.2.2 Upload and enrichment tools

The ``Uploading´´ step

Ones the oDBdata files are available at the OEM they are moved to Chalmers. The process of how the upload is performed is described in the flowchart below. This include the steps ofrom when the software is arriving at Chalmers from the OEM until it is uploaded into the database (and the files are moved to appropriate folders)
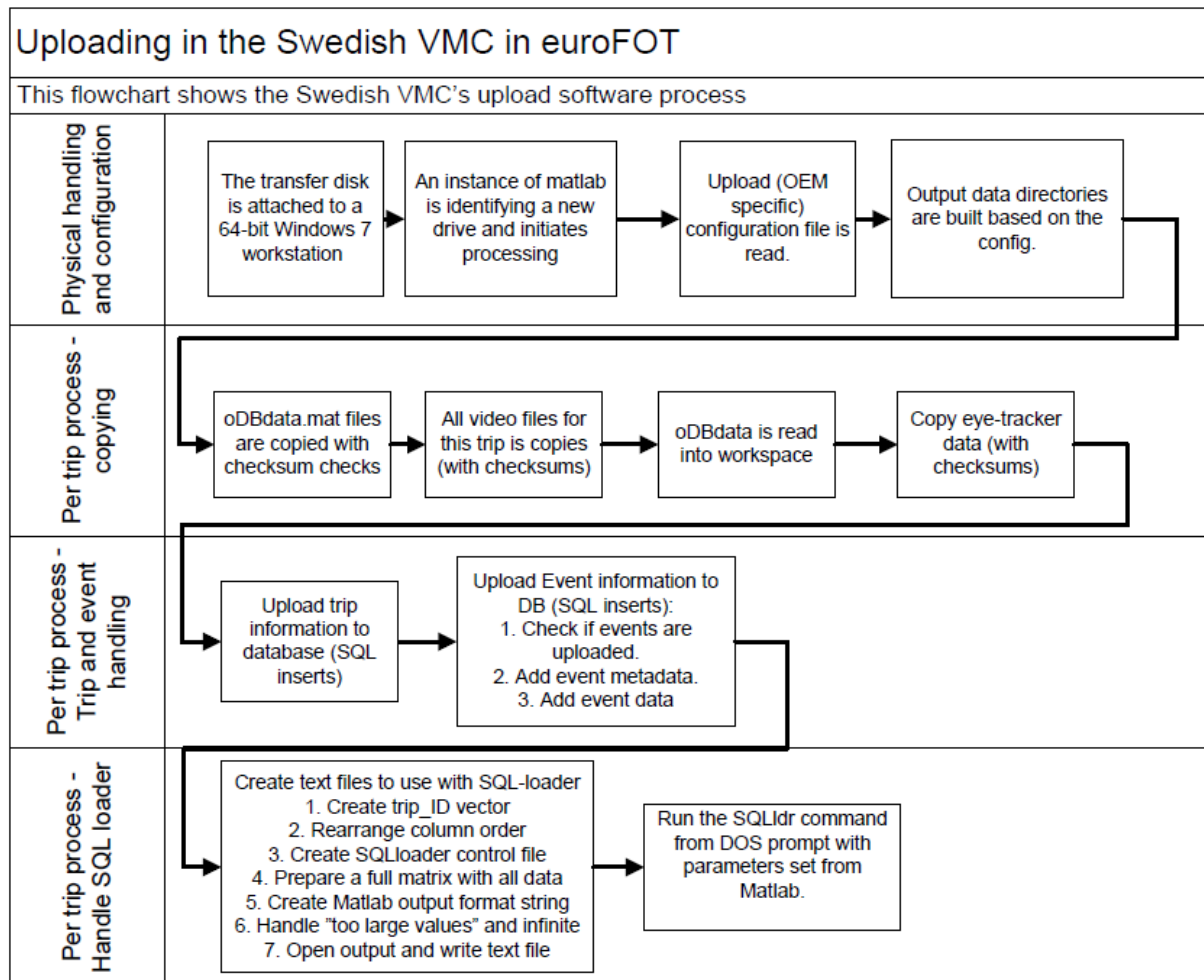

Figure 46: The uploading steps in the Swedish VMC.

NAVTEQ's ADASRP, a Windows-based framework application for prototyping ADAS and navigation solutions, is used to complement CAN and other sensor data from the vehicles with map data. This is done during the pre-processing of the logged data. A Volvo ADASRP plug-in, GPSToRoute, reads a file containing GPS positions (latitude and longitude) and then finds the most probable route given the NAVTEQ map (using Dijkstra's algorithm). Interesting map attributes along this route are then extracted or calculated and exported to a cvs file where every original position is followed by the corresponding map attributes.

The NAVTEQ map is divided into several hierarchies and the GPSToRoute plugin deals mostly with links. A link can be a road, several connected roads, or a part of a road. Each link consists of two or more shape points, which define the geometry of the link. The end points of the link are called start node and end node, and they are sometimes also called node 0 and node 1. Thus, when link direction is 1, it means the vehicle is travelling from node 0 to node 1. GPSToRoute is mostly concerned with attributes on the link level; for each raw GPS

position, it maps the corresponding link and extracts those attributes. For euroFOT, 60 map attributes are collected, including speed limit, road type, number of lanes and distance to next crossing.
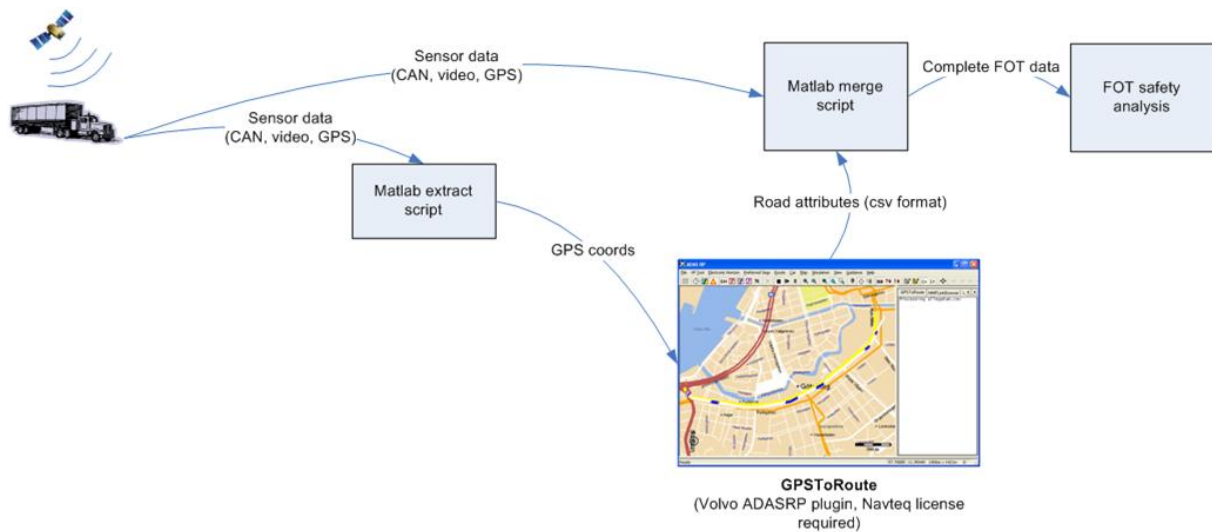

Figure 47: Overview of the process of complementing logged data with map attributes.

3.2.2.3 Lessons learned

Limitations and issues

- Data quality is difficult. Even if a lot of quality checks are made in the data management chain it is difficult to utilize this easily, e.g. what is poor quality for different types of analysis for different measures and events? How can it be made transparent to the users what quality measures to use and what they mean? How we proceed: The use of data quality will have to be in focus for the initial analysis and simple guidelines have to be written. Even if this has been done to some extent in the piloting, the main analysis is still ahead (and with that the understanding of use of quality).

- Implementation validation must be done iteratively. Lessons learned from previous studies have shown that manual review and evaluation of ALL measures and events before production upload should be done. This is something that has to be done several times and takes significant time, but it is crucial for the quality assurance and good analysis.

- Availability of map data. For several reasons the map matching tool and map data enrichment tool were available quite late in the project and thus were not integrated in the pre-processing at the beginning. Therefore, we have had to add a way to add map data on already uploaded data.

- Not possible to know all possible analysis, derived measures and events needed a priori. The piloting phase of the pre-processing and uploading included the full chain until test analysis and methodologically answering a few research questions, but since the amount of data needed for proper statistics and full analysis was (obviously) not available, it was difficult to plan for and implement all possible derived measures, events and quality checks before analysis. As a result of this, the analysts would have to review each hypothesis and revisit the implementations of derived measures and events, and probably implement several new or refined ones along the way during analysis.

Successes

- <u>A user friendly interface using spreadsheets for defining pre-processing significantly simplified usage and implementation.</u> The pre-processing and uploading was implemented so that adding new measures, derived measures, and events etc in the development phase was easy. That is, a framework was implemented to allow for only needing to add information about the measures in a XLS-spreadsheet, and when needed implement specific filters or handling criteria in MATLAB (with dynamic references in the spreadsheet). This document also made sure that all documentation was gathered in one place only.

- <u>Several layers of quality checks were implemented.</u> The solution we used includes a number of layers of data quality checks that is used and added to the database, making it possible for analysts to readily access and filter data with respect to quality on a per sample (on some measures), per measure and per trip basis. The quality checks were also implemented so that qualities was calculated in several tiers, allowing for more complex but still easily traceable quality checks.

- <u>One data format was used</u> throughout the pre-processing, uploading and analysis (based on MATLAB structures), once the raw data had been read. This simplified implementation aspects and application of functions and filters throughout the project. This format is complete enough to easily be able to transform the data into most other formats (if a parser/writer is written).

- <u>Hypothesis breakdown:</u> In order to be able to implement the piloting and prepare for main analysis, significant work has been done (by other work package) to break down the VMC specific hypothesis into specific Hypothesis Breakdown document (one per hypothesis, not all finalized yet). This has been a great help in the implementation of the pre-processing and piloting and assuring a solid link between analysis and data management.

Performance

<u>With respect to quality</u>

The quality checks are performed on per sample, per measure and per trip basis.

<u>Pre-processing data verification</u>

To ensure that the pre-processing in itself does not cause any significant alteration of the measure data, a data verification step has been added at the end of the process. This consists of a number of checks and comparisons between the processed data and the corresponding raw data, briefly described below:

- Time vectors are checked to ensure that all values are consistent, i.e. monotonously increasing and evenly spaced.

- Signal values are checked to determine if they are constant or consist of invalid (NaN) values only and that the raw and processed data are identical in such cases.

- Mean value and standard deviation of raw and processed data are compared. This test is passed if the mean value and standard deviation differs by less than 5% of the root mean square (RMS) value of the signal.

- By means of cross-correlation, signal similarity and potential time shift is determined. The test is passed if the cross-correlation coefficient is more than 0.9 and the time shift is less than one second.

- For enumerative type signals an additional check is performed to ensure that the number and values of the signal levels in raw and processed data are the same.

The results from the pre-processing data verification are stored in the segment info and the measure info in the data base structure. The verification has led to detection and correction of a couple of small but significant errors in the pre-processing. It is of course very important to resolve all problems early in order to avoid the need to rerun the pre-processing on very large amounts of data later. Currently, the data verification results show very good similarity between raw and processed data for all measures.

<u>Pre-processing quality check</u>

While the aim of the pre-processing data verification is to make sure that the data is not majorly altered in the pre processing, the aim of the pre processing quality check is to make sure the data coming from the sensors are of good quality.

This is done by comparing the signal to typical signal characteristics, for example check that the signal is within the sensor range or the natural range (limited by the laws of physics and other external factors), and creating a quality measure/value that is uploaded to the database. This quality measure can then be used to identify potential problems with the sensors and to help the analyst to exclude poor quality data.

As mentioned above there are three levels of quality checks: per sample, per measure and per trip:

1. Per sample

On this level of the quality check a quality value is created for each sample of the signal by checking that the signal value is within a reasonable range, and forming a per sample quality vector. This quality check is used for intermittent sensors (sensor that give no or unreliable data under certain conditions), in our case GPS, radar, lane tracker and eye tracker. For GPS it is checked that the receiver has satellite coverage at the time point the sample was taken and for radar and lane tracker, compound quality measures are formed from the CAN signals containing information about the availability of these systems.

For the eye tracker signals, the per sample check is performed in a slightly different way, and no per sample quality vector is formed. Instead a filtered version of the signal is created that contains all the values that can be considered to be of good quality. This is done by processing the eye tracker signals trough a weighted median filter that takes into account quality signals from the eye trackers as well as constraints on eye movements.

2. Per measure

In the per measure quality check a quality value for the whole signal is created. The per measure quality checks that are performed on the majority of the signals can be separated in to an enumerative signal check and an analog signal check, depending on the type of the signal.

For all signals that go through the analog check it is checked that the signal is not constant and that the max and min values are within the specified signal range. Additionally, for some signals thresholds on sample to sample variations, and percentage of time the signal can have a zero derivative are set up.

In the case of the enumerative check it is made sure that the signal values match up to the specified enumerative signal values, and if the data was collected during baseline and the system is supposed to be turned off during baseline it is checked that all the signal values corresponds to the off value. Also, in the case of for example the BLIS signal which is a warning it is checked so that it is not constantly giving a warning.

Except for the above mentioned per measure quality checks there are a few exceptions as for example for the GPS data, where the per measure quality is aggregated from the per sample quality vector, by setting a threshold on allowed data loss from this sensor. Another exception is the vehicle speed which basically goes through all the steps of the analog check mentioned above, but also if the per measure quality of the GPS speed is

good has to fulfil some requirements on time lag and minimum correlation with the vehicle speed. Further exceptions are the eye tracker signals that undergo per measure quality checks that are slightly different from above, here the per measure quality is more based on thresholds of minimum time the eye tracker data is available during the trip

3. Per trip quality check

The per trip quality is a tool for quick identification of trips that should be excluded from analysis and are based on thresholds of a minimum trip duration of 120 sec, minimum trip length of 100 m and that not more than 95 % of the trip has zero speed. Furthermore it is checked that the CAN source is available for the trip and that the per measure quality of the vehicle speed is good.

The quality measures from all of the above mentioned steps are presented in the segment info and the measure info in the data base structure. In the processing and quality of data from pilot data the following indicative data and processing quality have been found:

-   GPS data is generally available for all trips and there are reliable data around 95-99% of the time. Lack of data is mainly due to obstructed signals during tunnel passages etc.

-   CAN data is available more or less 100% of the time.

-   Eye-tracker data is available for most trips and there are reliable head position and gave angle data generally in the interval 30-70% of the time. This fairly large variation is probably due to differences e.g. in head position (in relation to the camera) for different drivers.

With respect to speed

One aspect of pre-processing is that it must take significantly shorter time to process than to record to be able to keep up with data delivery. This has been a guiding factor in implementation with focus on identifying bottlenecks. Many parts of the process are so fast in relation to the bottlenecks that no time is needed to be spent to optimize these parts. Also, due to the hardware availability for the pre-processing at the OEMs (large multi-core clusters), less time has been needed to spend on optimization.

In the pre-processing the following gives an indication of the relative time between the different tasks and as absolute values:

Tasks	Seconds per hour collected data (workstation)
Copying of data from transfer disk to server	54s/h
Decrypting data	54s/h
Reading data from raw format into MATLAB	20s/h
Decoding CAN and do minor processing	39s/h
Perform all filtering, resampling, quality measures, derived measures and events for everything except eye tracker	35s/h
Perform filtering and derived measure calculation for eye-tracker	70s/h
Perform Kalman filtering (position enhancement)	Included in step 5
Map matching	Parallel task
Unpack and convert video to X.264	Unpack: 138 s/h X264-conversion: ~2000s/h
Copy video data from transfer disk to server (eSATA with RAID0)	17s/h
Load oDBdata into workspace	<1s/h
Upload trip information to DB	5s/h
Upload event information to DB (will differ significantly between trips).	3s/h

Tasks	Seconds per hour collected data (workstation)
Create text files with data for SQLldr upload	22s/h
Uploading of data (text files) to DB with SQLldr	5s/h

All times are for one single process.
For fields marked yellow we have 3 parallel processes (that is, the processing time should be divided by three).
For fields marked green we have 16 parallel processes.
For fields marked turquoise we have 12 parallel processes
For fields marked pink we only have 1 process.
For fields marked red we have 1-6 processes, but manually started.

Future recommendations

- Overall the cluster environment using MATLAB parallel toolbox (Parallel Computing Toolbox - MATLAB 2011) has worked very well. The drawback has been that management of unique identities has become more complex.

- Separating the handling of incoming data from pre-processing has been a success. Especially since all steps in the pre-processing were not finished before data collection started.

- Evaluate at an early stage all the software needed. At the Swedish VMC some software packages (decoding CAN, NAVTEQ map data) needed Windows as operating system, causing a more complex automated flow on the Linux-based high performance computing (HPC).

- By converting video data from xvid (Home of the Xvid Codec 2011) to h264 (H.264/MPEG-4 AVC - Wikipedia, the free encyclopedia 2011) the data storage need was reduced by 40% at Swedish VMC. The process is very CPU intensive and a cluster of computer nodes were needed to perform this task.

- The large amounts of video data exceeded the filesystem limits of storage appliances. Therefore a server providing logic to be able to create a single mount point was needed. For future projects different filesystems and appliances should be evaluated; for instance by using ZFS (ZFS - Wikipedia, the free encyclopedia 2011) or equivalent for almost unlimited shares.

- A 'sandbox' database including a subset of the data should be created. When developing and refining scripts it is much more time efficient to test on a smaller dataset. When satisfied the analysis can be executed on the whole dataset.

- To be able to use SQL for analysis on billions of rows in a single table effectively, indexes needs to be created. It is also vital to consider the storage needed for all indexes.

- Use and analyse explain plans before executing queries. By doing this many bad written SQL can be avoided.

- Try to fully use the power of state-of-the-art servers utilizing many cores. By using parallel techniques in the database server or splitting the queries into smaller pieces can boost the performance.

- Consider the cost for making backups of the database. If raw or pre processed data is stored and has backup there might not be need for taking backups of the continuous

data tables. The time for re-uploading data might take longer time but it is important to state the service level agreement for downtime.

- There is a great cost (in terms of computing time) for performing joins when dealing with large tables. By extensive de-normalization of the schema and even duplicating data analysis can be faster.

- Storing large amounts of data in a traditional relational database should be questioned for future projects of this scale or bigger. The trend is that more specialized software is used for corner case purposes. New techniques as columnar databases, Hadoop (Welcome to Apache Hadoop! 2011) including HBase (HBase - HBase Home 2011) and Hive (Welcome to Hive! 2011) are making progress and should be evaluated. Also new products for hardware and software acceleration could be considered for better performance. It is also an upcoming challenge to see how different storage and analysis software can be used to complement each other. By mastering this researchers can be much more efficient in analysis projects and by using the right tools at the right time new methods for analysis projects can be developed.

- For pre processing and analysis these large amounts of data requires 64 bit memory structures and software support.

- Data access shall be role based.

- Data access on filesystem is not as flexible and granular as in databases.

- Network capability is important to consider when handling large amounts of data. At the Swedish VMC, copying data in parallel from three USB-disks to storage exceeded 100mbit connection and 1gbit was needed.

- All data sent between OEM and the Swedish VMC was encrypted using Truecrypt (TrueCrypt, 2011) ensuring the sensitive data.

3.2.3   German-2 OC

The data logging of Daimler and BWM contains CAN-data which are converted to even-spaced time series data with 10Hz, and video data with 5Hz. The data are logged in the vehicles in data formats that are specific for the OEM. The OEMs are responsible for converting the logged raw data into even spaced time series data in .mat format (MATLAB format).

The converted CAN-data and the video data are copied onto 2TB external hard drives. Whenever a hard drive is full, the OEM sends it via mail to the University of Würzburg. Here, data pre-processing and upload to the database take place.

The server for the German-2 OC is installed at the location of WIVW GmbH in Veitshöchheim and it contains data from BMW and Daimler.

3.2.3.1 Data storage solution

Data backup routines

Raw data. Backups for logged raw data as well as for video data are run at BMW and Daimler. At IZVW, two copies of the raw data are stored on external hard discs.

Pre-processed time series data. The pre-processed CAN-data is stored as mat-files on the server at IZVW. There are 2 hard drives with 2TB each. One of them is used for BMW data and the other one for Daimler data. For both internal hard drives of the server, a backup is run every weekend.

Other data. This section relates to the following types of data:
- description of the data structure
- mat-structures containing the used variable names
- MATLAB-routines for uploading, pre-processing and analysing the data
- MATLAB-routines for analysing the data
- other routines (e.g. for data conversion, extraction of map based data)
- overview tables generate in the process of data analysis
- tables containing aggregated data

The named files are added to the version control program Tortoise (tortoisesvn.tigris.org 2011). Here, backup routines are run at WIVW on a daily basis.

Furthermore, the weekly backup for the internal hard drives of the server also covers the above mentioned files.

Description of Database-Structure

The overall data structure used for the German-2 OC is displayed in Figure 48. The data can be separated into subjective data collected via questionnaires and objective data logged in the vehicles. The subjective data is stored in an Access-db. From the objective data, an overview table that lists relevant information per trip is created. The overview table is integrated into the Access-db. Through that, a direct link between objective and subjective data is possible.
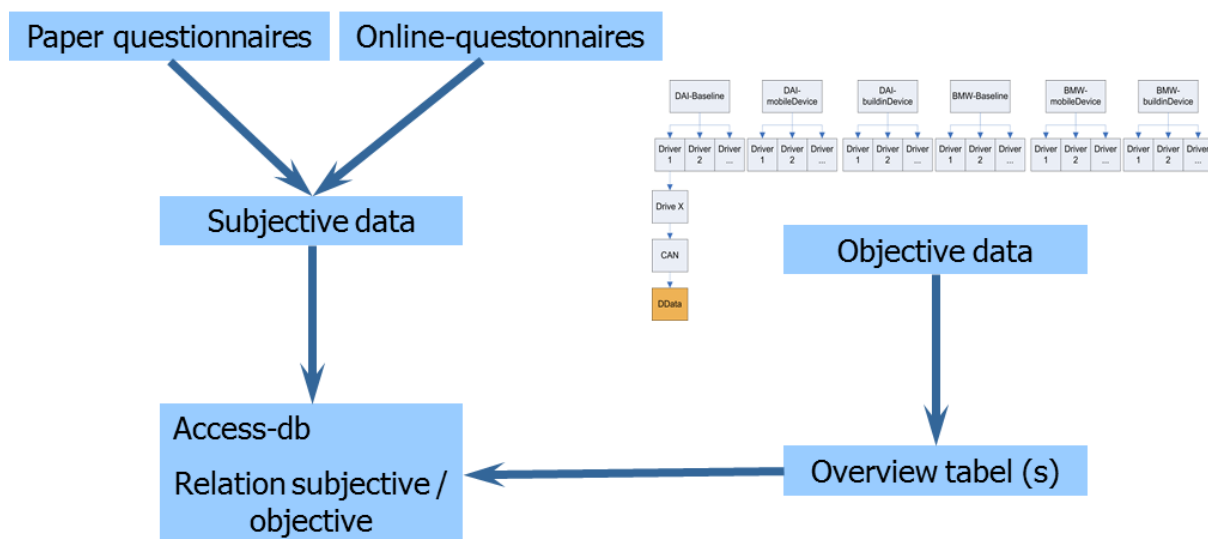

Figure 48: General layout of German-2 OC database.

Structure of objective data
At the German-2 OC, a defined folder structure is used that contains one mat-file per trip. The used structure relates each trip to a condition and a driver-ID. A similar structure is used for storing the video data on the external hard drives. The used structure allows to link for each trip the mat-file stored on the server to the video stored on an external hard drive.
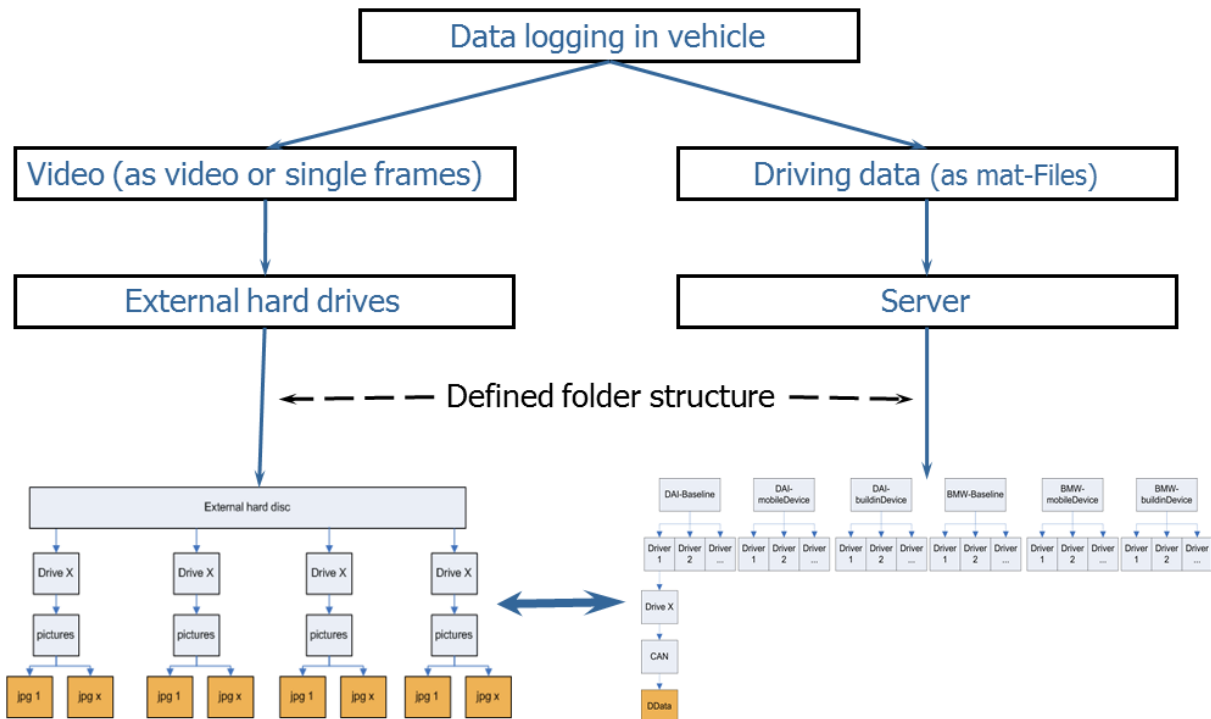
Figure 49: Adapted folder structure required for CAN-data stored as .mat-files

The used structure supports a division of the data into is conditions. Each driver-ID can occur in each condition once. The mat-files in the different conditions can differ in number of variables.
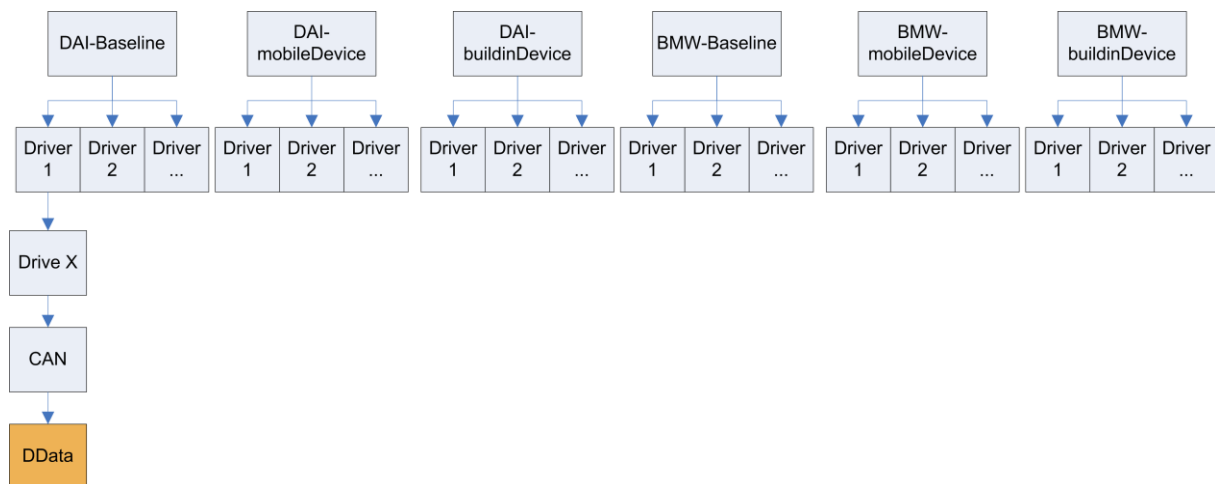
The following structure is used in the German-2 OC:

Figure 50: Folder structure used for storing the driving data.

Video data

For both OEMs, data logging does not only include CAN but also video data. The formats used for video differ between the OEMs.

BMW saves the video data as single frames in jpg-format. The naming of the frames allows relating each frame to a certain data point in the driving data.

For Daimler, the logged data can be converted either to single frames or to videos. It has been decided that using video format is handier and therefore, the data is stored as videos. Daimler converts the video data into videos in mjpeg-format. For longer drives, the whole video is stored in several files with a length of about 50 minutes each. At IZVW, the different

video sections for one trip are merged into one mjpeg-video which is stored in a Matroska-container (Matroska Media Container - Homepage, 2011).

3.2.3.2 Upload and enrichment tools

Description of data

The variables of the logged CAN-data used for euro-FOT as well as all derived and added parameters are described in detail in a file called Var_db.xls. The variables and also the number of variables differ between BMW and Daimler. In Var_db the following information is given for each variable:

- name and source (e.g. raw log data, map data)

- unit, resolution and range for continuous variables

- meaning of categories for categorical data

Steps of data processing & enrichment

The pre-processing routines are organized in a way that they always work on all files in a specific folder, write the processed files into a new folder and delete the input files. Through that it is ensured, that each step is run only once on each file. At the end of the data pre-processing routines the files are stored in the structure described above. Figure 51 gives an overview over the different steps of data pre-processing, enrichment and upload.
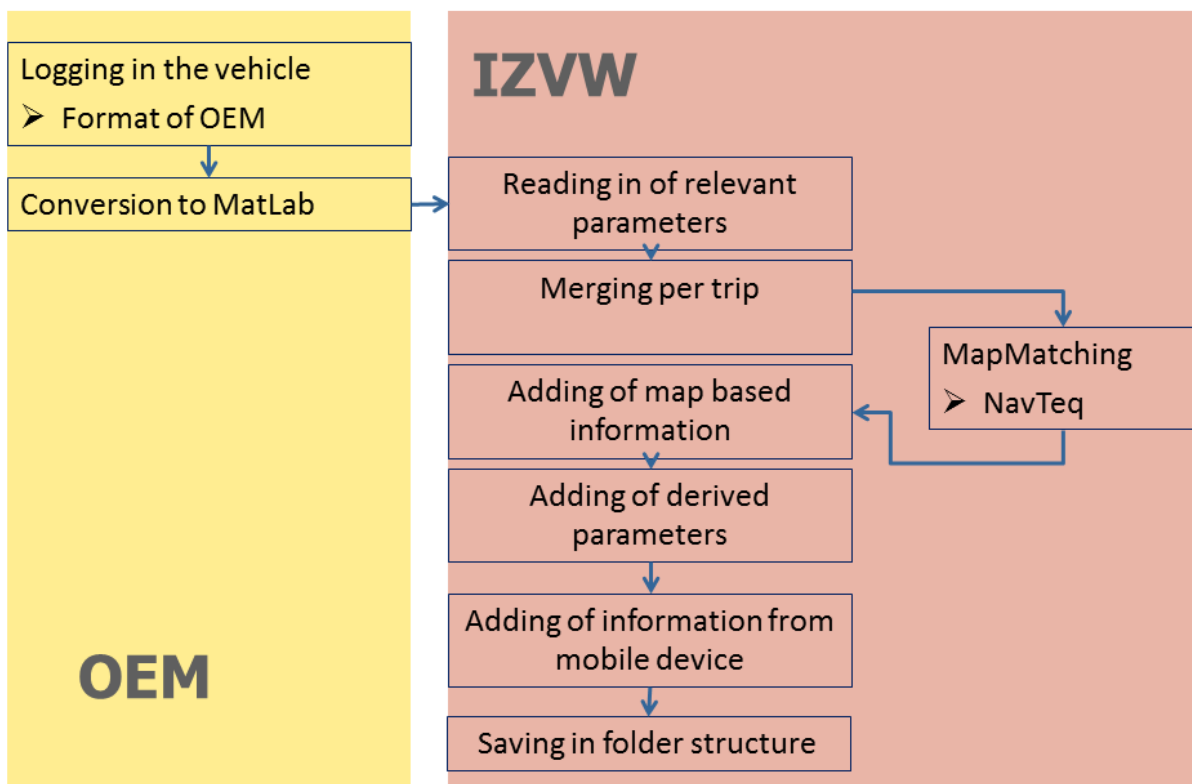

Figure 51: Overview of data pre-processing, enrichment and upload routines.

- Logging in the vehicle: the data is logged in a format that can only be read by the OEM

- Conversion to MATLAB: The logged raw data is converted into even spaced time series data that is saved in mat-format. This step is done at the OEMs.

- Reading in of relevant parameters: Data logging includes more signals than needed for the euroFOT analysis; only signals needed for euroFOT are kept for the db. If necessary, missing values are interpolated at this step.

- Merging per trip: Due to peculiarities of the data logging, it is possible that one trip is logged in more than one data-file. Files belonging to one trip are merged.

- Map matching: The NAVTEQ-tool is used to derive information from the digital map based on GPS-position.

- Adding of map based information: The parameters derived from the map are merged with the vehicle data per trip.

- Adding of derived parameters: Algorithms have been implemented that derive new parameters from the logged raw signals. The algorithms include adding of continuous signals (e.g. standard deviation of lane position), of categorical signals (e.g. weather based on wiper status and temperature) and of event detection (e.g. incidents). The derived parameters are added as new variables to each trip.

- Adding of information from mobile device: In the condition mobile device, the information logged on the mobile device is added as new variables to the vehicle data whenever the mobile device has been active.

- Saving in folder structure: The data consists now of one file per trip that contains all the parameters needed for the euroFOT analysis. The files are stored in the final folder structure that allocates driver ID and condition to each trip. Before storing the data in the final structure, an overview table is created that gives basic information for each trip.

3.2.3.3 Lessons learned

- The whole chain of data pre-processing, upload and analysis should be tested (if possible) until PI-calculation before the FOT. Some problems with signals (e.g. strange values) were realized very late in the whole process.

- The data pre-processing and upload procedures should be flexible and allow changes after the FOT has started: This is necessary because new problems might occur that did not exist in the pilot. For instance, in the Daimler fleet vehicles data logging continued sometimes for hours after the end of a drive. As a consequence, more than one drive were often logged in one data log. This has not occurred in the pilots, because due to an error in the logger installation, the shut-down procedure did not work properly in the fleet vehicles but it worked well in the pilot car. It took some months to realize and find the error and then to change the cabling in all fleet vehicles. Because of the error, a new step had to be included in the data pre-processing procedures that now separates each data log into single drives.

- Data handling (especially of video data) should be tested with a large amount of data already in the pilots: In the beginning, also the Daimler video data was stored as single frames instead of video format. Although there were a large number of separate files for each trip, the handling of the videos (e.g. copying) was no problem with the pilot data (about 50 hours / 3000km of data logging). After converting the first months of FOT data, it was realized that because of the huge number of single files, copying the data from Daimler conversion in ext3 (ext3 - Wikipedia, the free encyclopedia, 2011) to IZVW server in NTFS (NTFS.com File System Structure, Recovery Software, Hard Disk Internals, 2011) file format took longer than measurement time. As a consequence, the format had to be changed from single frames to video format and the database and data analysis tools had to be adapted. Especially the adaptation of the video tool took a lot of effort. Now the tool works with both, single frames and video format.

3.2.4   Italian VMC

3.2.4.1 Data storage and upload solution

Since the Italian VMC only collects questionnaires data (see also Section 2.2.5), the storage solution and upload procedure used are much simpler compared to those used in other VMCs.

Driver's background data, timing of handing-out and receiving-back of filled-in questionnaires, incentive scheduling and timing, and raw file of filled in questionnaires are stored in MySQL-PHP database.

The questionnaire data is collected in two ways:

1) electronically based, by using an open source tool LimeSurvey (LimeSurvey - the free & open source survey software tool! 2011) and

2) paper based. The paper based questionnaire data is first scanned and converted into digital format and then imported into LimeSurvey tool by manual data entry procedure.

3.2.4.2 Lessons learned

The main lessons learned from the Italian FOT, in terms of management of questionnaire data, are:

1. It is very important to develop and improve a web-based survey tool for data collection in order to save time and prevent data transcription mistakes.
2. In order to reach more drivers and improve response rate, it is very good to duplicate or offer multiple options for filling questionnaires in (i.e. hard copy and electronic copy).

4   Data analysis tool

Data analysis in euroFOT is supported by the tool developed by WP3500. The tool is an application with graphical user interface (GUI) aimed for browsing, replaying, visualising and annotating recorded FOT data over time for one trip at a time.

The purpose of this chapter is to describe the results for the base software for analysis of field operational vehicle data.

New data collection abilities created exciting scientific possibilities and analysis challenges due to the scale and richness of data that is collected. For example, coding and careful analysis of video data remains extremely time-consuming. Other forms of data, such as sensor readings, while useful as separate data sources, may provide more valuable insights about activity when viewed and analysed together. Visualisation tools are needed to address the challenge of analysing and synthesising information from multiple data sources and at multiple time scales. Sharing the content of the user findings and communicate it, is another fundamental issue in the analysis tools.

Although several other systems have been developed to support various aspects of this visualisation challenge, data analysis software represents a unique focus on the visualisation of multiple diverse data sources. For example, numerous systems exist for coding and annotation of video, such as VCode (Hagedorn, Hailpern, & Karahalios, 2008), Diver (Pea, Mills, Rosen, Dauber, Effelsberg, & Hoffert, 2004), ELAN (Wittenburg, Brugman, Russel, Klassmann, & Sloetjes, 2006) and The Observer XT (Zimmerman, Bolhuis, Willemsen, Meyer, & Noldus, 2009). While these systems have powerful coding capabilities, they are not designed for analysis or visualisation of heterogeneous data, nor do they support easily extensible automatic analysis facilities. They are also commercial products with limited open extensibility to visualise new types of data.

The main contribution from WP3500 is data analysis software that allows researchers to visualise time-based data from multiple sources and manually or automatically annotate the data.

The main design goals that motivated the work in this package are:

1. The tool should be easy to use, with the interface facilitating the annotation process;

2. It should be flexible and not limited to specific research questions, allowing users to define annotation categories;

3. It should be expandable so that the users could implement new modules and functions;

4. It should provide users with an interactive timeline-based visualisation of observational data;

5. It should be able to efficiently retrieve information from both file management systems and relational databases.

The development of this software is completed in spring 2011.

4.1   Method/Procedure Used

The development of the data analysis tool involved defining the use cases and requirements, development of the technical design, implementation and testing of the data analysis tool.

Common templates were used as a base for gathering requirements and use cases by all VMCs. The templates, described in Deliverable D3.2-Annex 3 (Kovaceva, et al. 2011), gave an efficient structure for unifying use cases collected from each VMC, which were then used to derive requirements for the VMCs partners. The requirements were categorised into high priority (i.e., common for all VMCs partners) and low priority (i.e., specific for different

partners). For example, each VMC had specific low priority requirements, due to differences in sets of safety functions, sets of hypotheses and constraints, methods of data collection and data type, database implementations and upload tools and data analysis plans.

After definition of the requirements, the common software architecture was designed. The architectural design refers to the broad functions, high-level abstraction, of the system.

General principles were adhered in the design and coding of the data analysis tool, testing, help and support, software releases and software reuse.

At the end the base data analysis tools were implemented in each of the VMCs. The tools have some common functionalities as well as some VMC-specific functionalities. Detailed descriptions of the data analysis tools developed in each VMC are provided in Section 4.2.

4.2 Base of Data Analysis Tools Used in euroFOT

Detailed descriptions of the data analysis tools developed in each VMC are provided in Sections 4.2.1 to 4.2.4.

4.2.1   German-1 OC and French VMC

French VMC and German-1 OC share a similar framework for data analysis software, although implementation varies a little between the two VMCs. The variations appear in (1) the implementation of the visualisation of the data, since German-1 OC is not using the video display option; and (2) the implementation of the enrichment scripts since the both VMCs have different PIs and set of events that need to be detected and characterised.

The aim of the data analysis software is to allow the definition and calculation of the intermediary results as well as their storage in the database, in the most interactive way. The software allows both visualisation of data (trip visualisation and annotation tool) and modification of the data model (definition of new PIs or events, linking of new scripts with trip data model front-end).

This software is mainly programmed in MATLAB.

Four processes are taking place in the workflow of the data:

- Pre-processing and enrichment – Raw trips are converted to a standard readable format (MATLAB file) initially only containing unsynchronised time-history signals. The signals are then synchronised and harmonised according to a uniform data model. The trips are then enriched with aggregated data, by applying user-defined scripts.

- Upload to database – Trips data from the MATLAB files is then stored in relational database.

- Analysis of data - The analysis is performed by directly querying aggregated data and its description from the database using standard statistical tools, such as SAS (Statistical Analysis Software, SAS/STAT 2011).

Users manipulate trip data only by:

- Defining new objects in the data model.

- Defining algorithms in scripts.

- Directly manipulating trip-related data in trip data viewer GUI, explained in the following section.

4.2.1.1 Implementation

The GUI consists of three main parts:

1) The Trips explorer, a search and listing tool,

2) The Trip visualisation and Annotation tool, and

3) Trip data model front-end tool also known as Experiment Builder.

Using the tools the analysts can:

- Search for trips corresponding to a specific set of filters, get information about processing progress, warnings and errors.

- View both raw (un-synchronised) data just uploaded from the DAS and synchronised data (extracted from the database).

- Modify any editable authorised attributes (especially in relation with events), and upload modified trip to the database.

- Prototype and test modifications of algorithms or data model, and push them in the production environment.

Trips Explorer

The Trips Explorer is a very powerful tool to search for specific trips, using filters, and access the corresponding data or test some algorithms on them.

Filters are based on standard status information for each trip (configuration, date, origin, error or warning), but also on any kind of data which has been declared in the Trip Data Model and created by the user during the analysis. It is therefore possible to search for trips where general attributes meet certain requirements, or which contain a specific kind of event, where an event attribute has such value. Filters are defined using the GUI pictured below, whose controls are entirely based on the data model. Filters can be saved and imported. Search is performed either by querying the database server, or using a simpler local (file-based) database, which is a transparently maintained replicate of some tables of the complete database. In the later case, the most complex search options are disabled.

Figure 52: Trips Explorer.

The Trips Explorer also integrates access to trip related data: either actual trip data (in Trip Data Viewer), or processing logs, in a dedicated visualisation tool (pictured below).

Figure 53: Log visualization integrated in Trips Explorer.

Finally, when the Framework is used in prototyping mode, the Trips Explorer allows running the processing on a set of selected trips, and testing data model or algorithms evolution.

Visualisation and annotation tool

This tool allows simultaneous visualisation of all trip-linked data. Data can be browsed and visualised using the GUI called *Trip Viewer*, shown in Figure 54. The trip viewer is the main module that controls all functionalities. The user can alter between the tabs in the panel on the left of the window. Each tab represents a separate functionality: Signals, Events & Situations, and Trip information.

In the first mode, the user can browse through the trip signals, which are listed in a table (see the left part of Figure 54). With the check boxes in front of each signal the user selects the signals that should be displayed as a time history on time graph (see the right side of Figure 54). The user can choose the style (graphical properties) of the line for each of the signals on the plot. The values of the signals in the current time (shown as red line on the time history graph, on Figure 54) can also be displayed in the module *Inspector*, Figure 55.
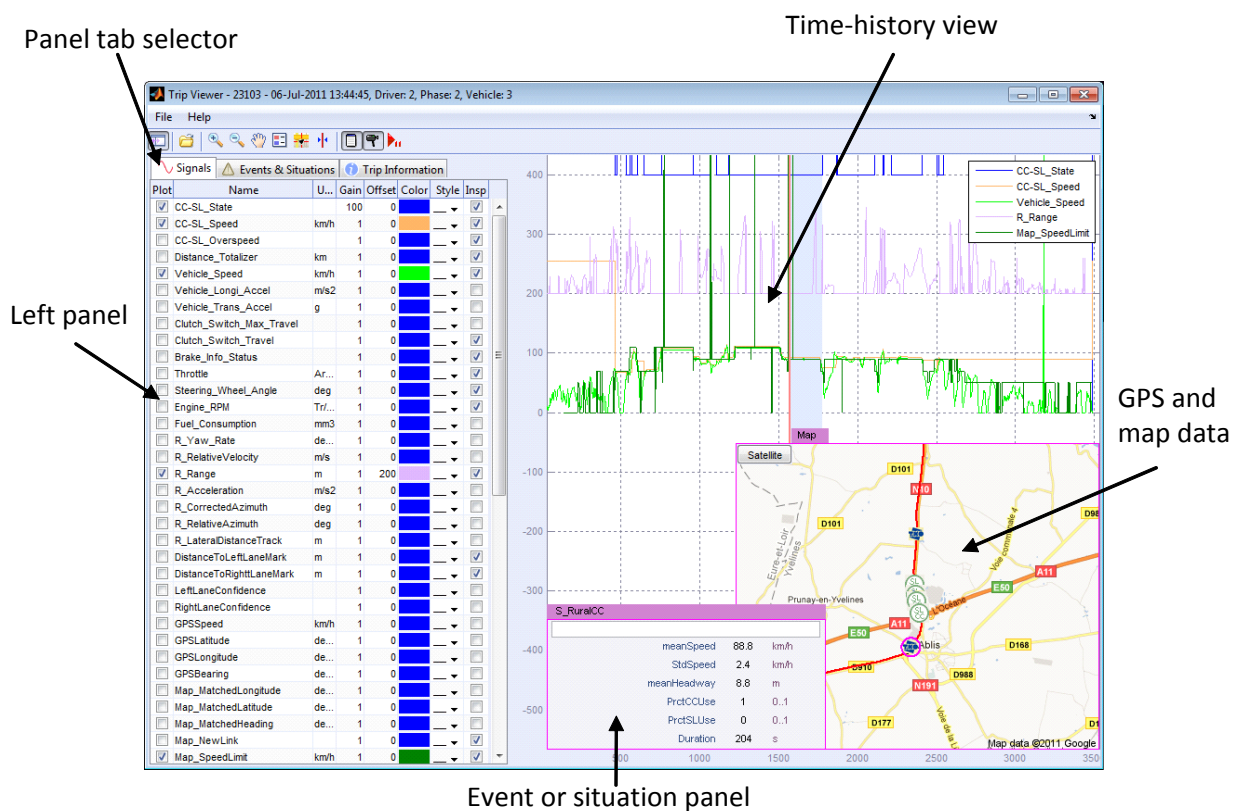

Figure 54: Trip Viewer - Visualisation and annotation tool, with signals selection panel.
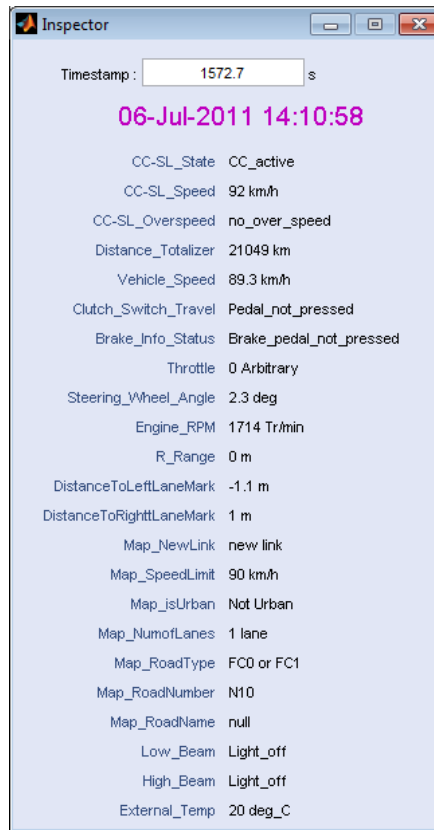
Figure 55: Inspector, showing the current values of the selected signals.

The user changes the current time by dragging the red cursor, entering a value in the inspector, or selecting an event or situation, in which case current time is set to the beginning of the instance.

Left panel in Events/Situations mode   Left panel in Trip Information mode

Left panel recessed
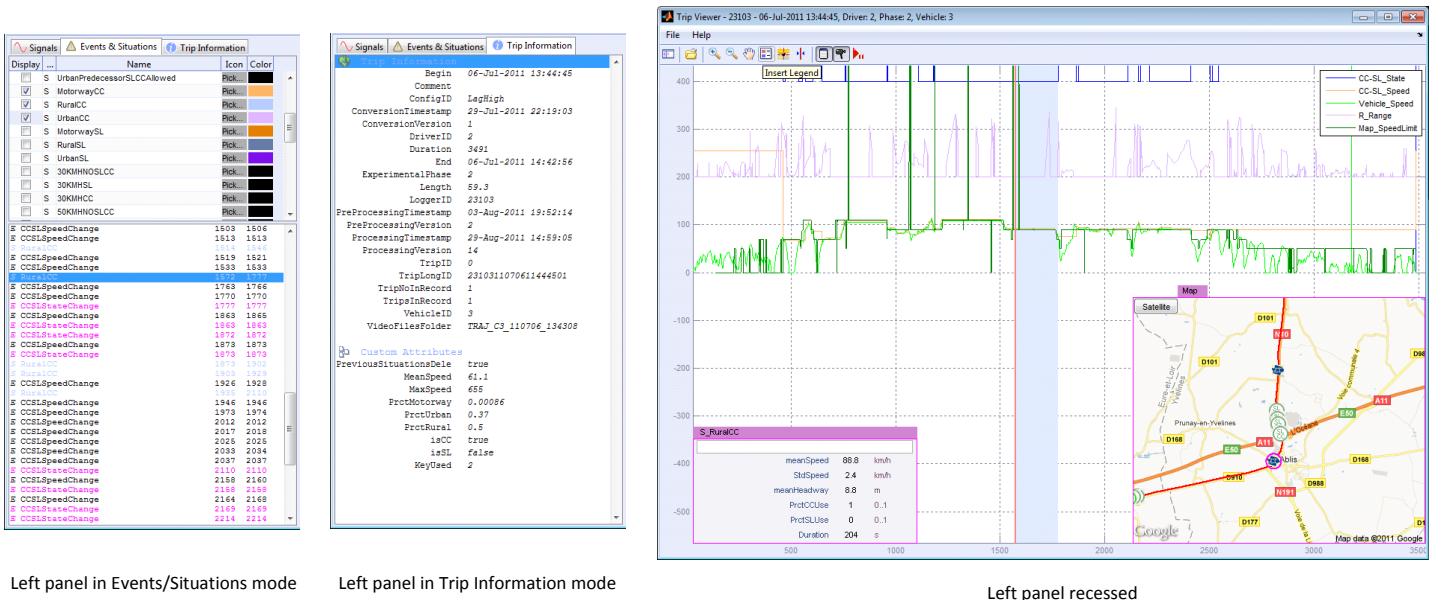
Each time the current time is modified, all information on screen (signal values, videos, vehicle's position on the map) is instantly refreshed.

When the user switches to the Event & Situations mode of the left panel, he is presented with a tool to filter the events and situations he wishes to display in a list and as icons on the map view.

When he uses the third mode, the user accesses general information about the trip, i.e. standard status, computed trip attributes, driver and vehicle attributes.

When using smaller screens, such as laptops, it is also possible to recess the left panel in order to leave more room for the actual data.

The video display module, Figure 56 displays the four camera views which are recorded by the French VMC: forward, driver, steering column and feet.

Figure 56: Video display; four camera views (Forward, Driver, Steering Column and Feet).

Images displayed in the video module are perfectly synced with numeric data. Keyframes caching when loading a trip allows extremely quick seek mode when changing current time by dragging the vertical red line. When the real-time play is activated, the current time in the whole GUI is constantly refreshed: current values in the inspectors are updated, as well as vehicle's position on map, and when necessary, map and/or time-history view are automatically panned.

The vehicle position is displayed over a map (Figure 57). The complete trip is lined in red, the vehicle's position is shown as a black triangle, and events and situations are represented as user-selectable icons. The user can change at will the zoom level and position of the map, by selecting the same "Zoom" or "Pan" control as those used for time-history data. Clicking on an event or situation icon has the same result as selecting the instance in the left panel's list: the corresponding event or situation form panel is displayed (Figure 58).

Figure 57: Vehicle's position on a map.

Events can be detected for each trip either by an automatic script, based on continuous variables or by manual annotation, using the event form. The manual annotation is done when the driver self-reports them, or when an operator discovers them from the video. Based on the video, the signals and operators own experience, the operator can discard an event (uncheck the check box *Valid Event*) in case the automatic detection was a false-positive. He can also annotate it with comments, in the text field, and modify some editable attributes, for instance by manually selecting values on a graph display. After each modification, the event's attributes, which are calculated and dependent on some of the edited attributes, are refreshed by the framework and the results are displayed in the form. This manual observation and validation of the detected events also allows troubleshooting and refining of the detection and characterisation algorithms.

Figure 58: Event module.

Trip data-model front-end tool

This tool allows performing all modifications in the data model (e.g. adding a new variable, defining a new script). It supports definition or modification of a new object in the trip data model: signal, situation, event and trip attribute by using the *Experiment Builder*, shown on Figure 59. The experiment builder calls the enrichment manager which runs a successive set of processes. A process is defined by its inputs, its outputs, and a script which computes the

outputs from the inputs. The analyst can define the processes by using the *'explore struct'* shown on the left panel of the experiment builder. The enrichment manager enriches and refreshes the trips by building a dependency graph, Figure 60, from the processes' definition, and by running the necessary scripts. The scripts are managed in a version control system (VCS). Local execution of refresher on currently loaded trip allows the analyst to preview the effects of a newly implemented modification on the trip data-model. The act of saving the modified data-model back to the database is done by the SQL interface which updates metadata tables and prepares data tables to receive newly computed attributes.
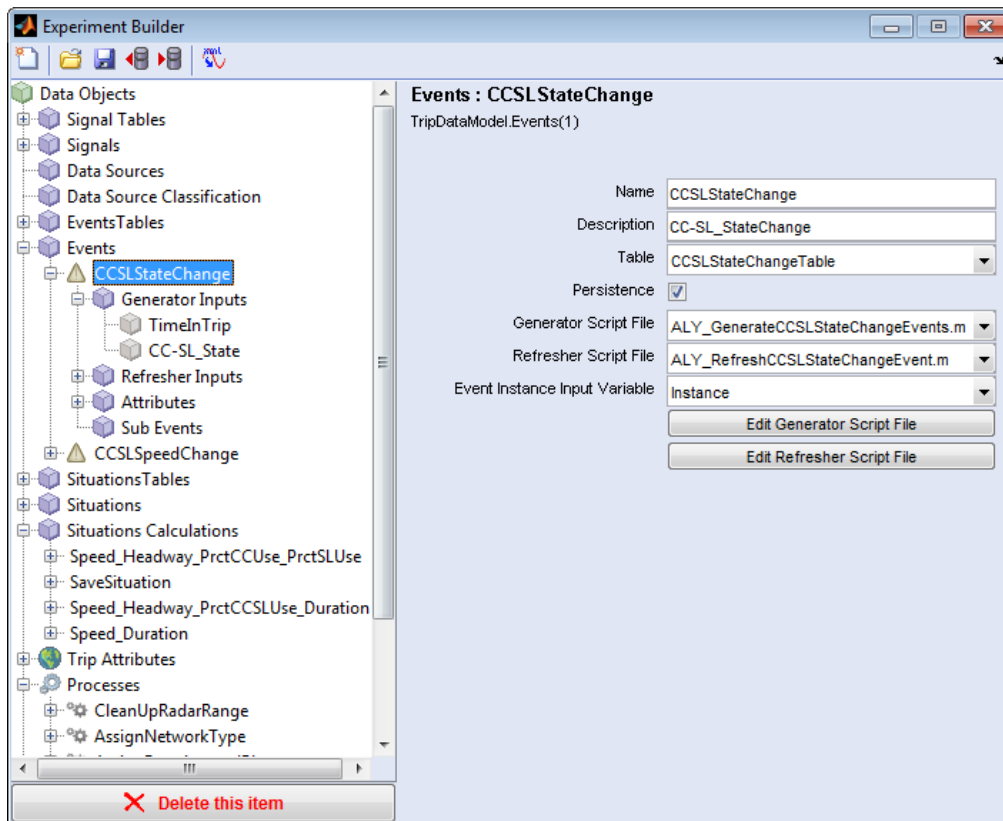

Figure 59: Experiment builder.
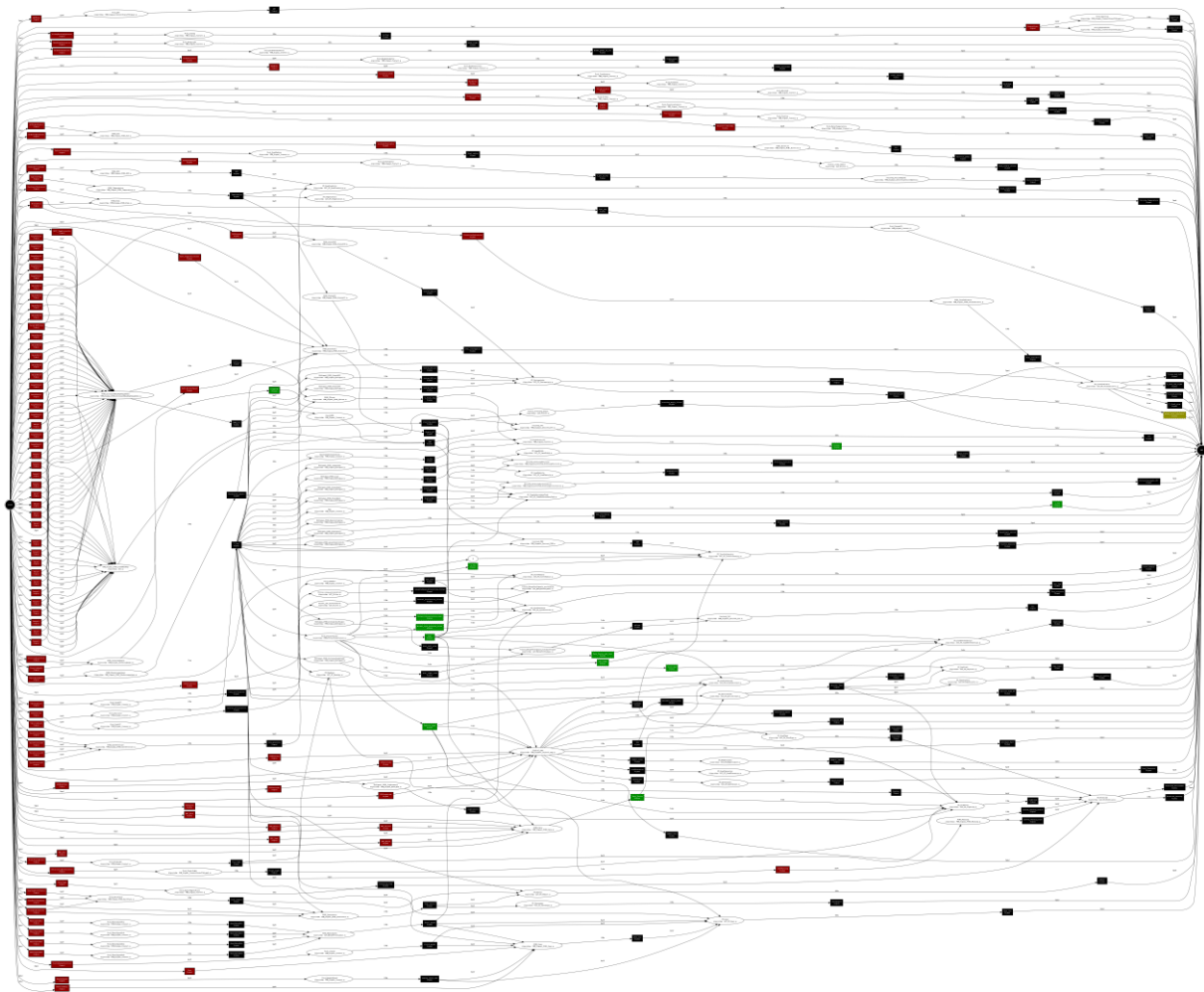
Figure 60: Dependency graph.

4.2.1.2 Tests performed and quality assurance

Release Process and Version Management

Releases are defined beforehand. Release and features are defined in the development roadmap. The releases are signed with a version number. The version number is a group of three numbers AA.BB.CC, where AA is the major release, BB is the Minor release and CC is a revision ID.

The roadmap for the releases foresees the usage of the numbers following to:

0:	Pre-Release FOT preparation
1:	FOT startup; software evaluation
2:	FOT operation; corrections and feature extension
3:	Data Evaluation; respecting results gathered from SP6
4:	Final release version

To ensure software quality, all software components and features are tested by the developers before releasing.

To keep track of software changes, a version management system is used. The version management system is based on Sub-Version. For the development workflow a strategy on how to handle different development paths in the version management system in parallel was defined.
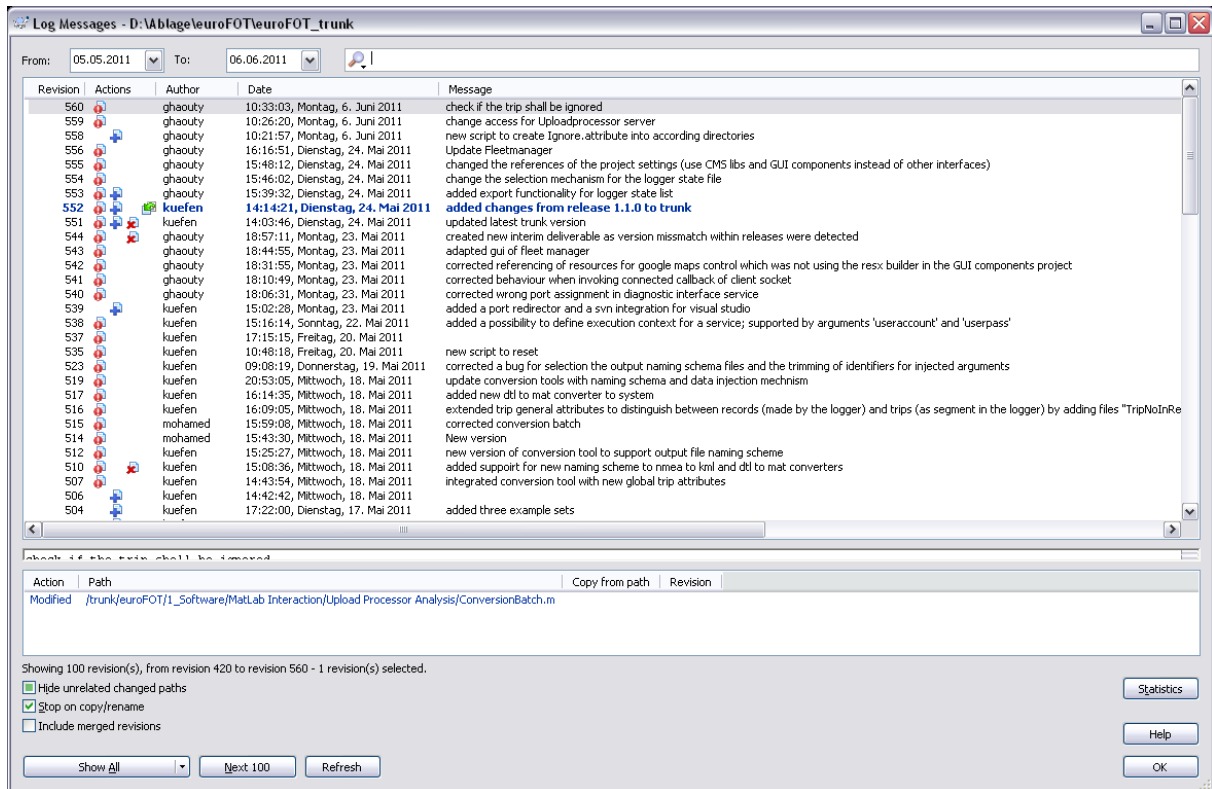
Figure 61: Subversion Changelog.

The main development is executed in the so called "trunk". For releases and corrections of the software within a release, the "branches" are used. For each release of a software component a corresponding development branch is created, which is marked by a "tag" before opening, when merging back to the trunk and after closing a release. The nomenclature of the tags is defined by "PREP_{releaseID}", "MERGE_{Cycle}_{releaseID}" and "POST_{releaseID}". The following graph shows the preferred usage of trunk, branches and tags.
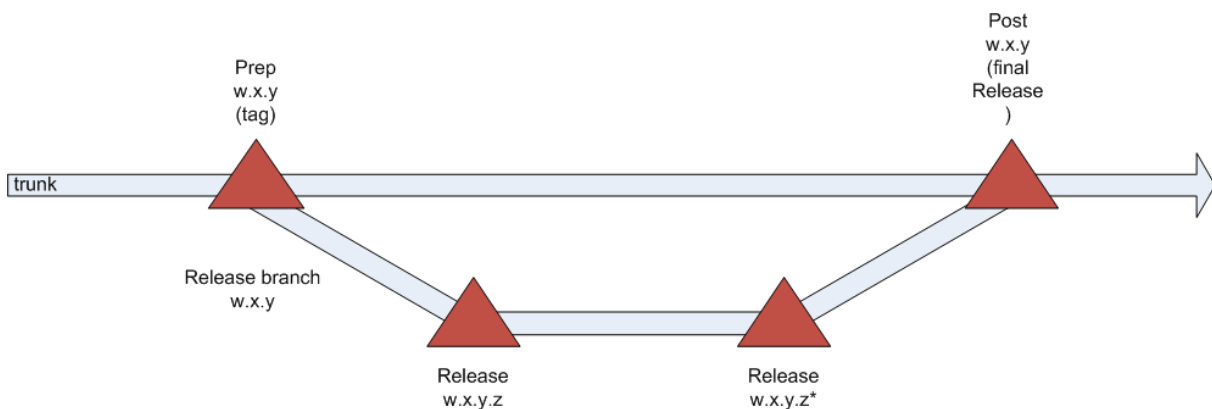
Figure 62: Branch Usage of Version Management.

Release versions of the software are accessible through the version management system at a predefined location ("Deliverables") for each software component. For each release, an individual branch of the "Deliverables" folder is created. Software updates can be gathered through this folder.

All binary files have a defined version number, which is created by a group of four numbers: AA.BB.CC.DD, where AA is the corresponding major release, BB is the subversion revision

number, CC is a continuously incremented number and DD defines year and day within the year of the build time. This versioning ID shall guarantee traceability of origin, creation time and corresponding sources of the binaries.

Help and support on usage, extensions and corrections of the software is provided by the developers of the software packages throughout the entire FOT operation time. Processes have been defined to ensure proper documentation and traceability.

To request additional features and to correct probable software errors, a Bugzilla based bug-tracking system is used (Home:: Bugzilla:: bugzilla.org, 2011). All users of the software have access to this bug-tracking portal and can report their issues and/or recommendations. Reporting includes the definition of the software component which is referred to as well as detailed information on environment, scenario etc. A sample report is shown below.
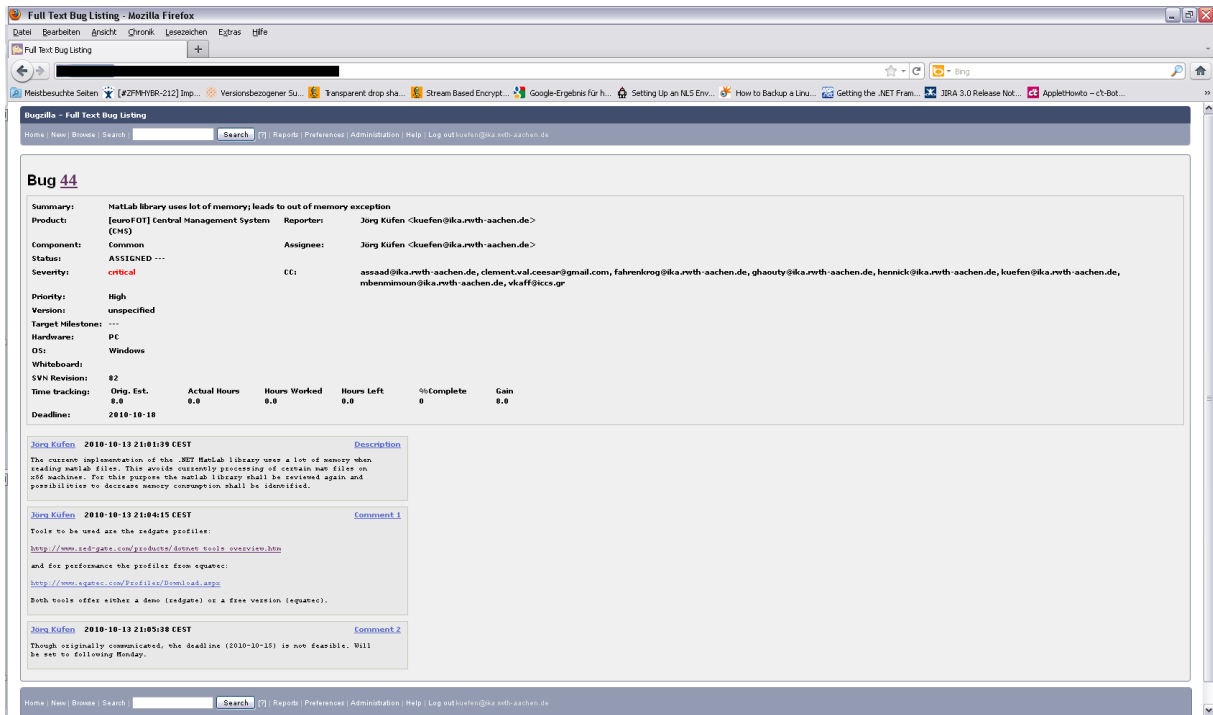
Figure 63: Sample Report for a reported Bug.

All reported features and requests are thereby documented and can be either directly be handled or be scheduled for future release versions. For software errors (bugs) and software extensions/changes different workflows and state schemas are used. The process state graphs for the processes "Software Correction/Bug", "Feature/Functional Change" and "Improvement" are shown below.
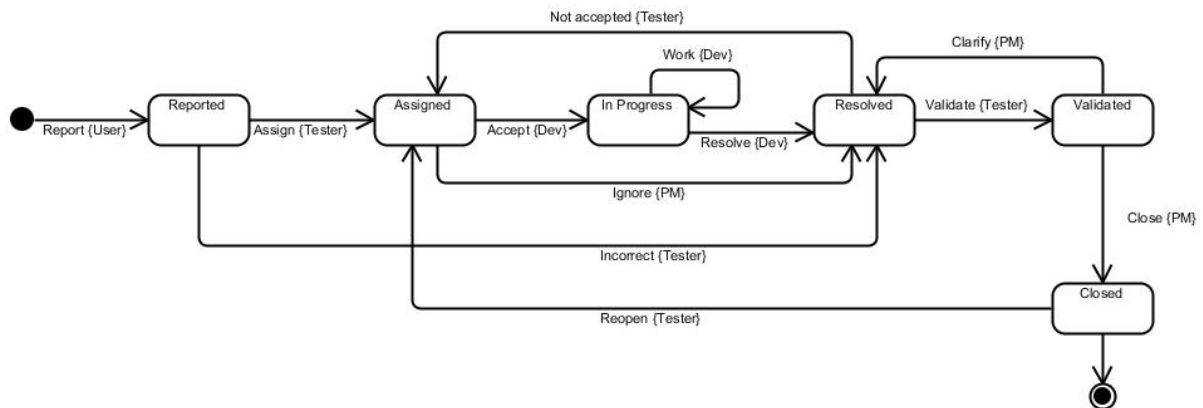
Figure 64: Software error/bug.
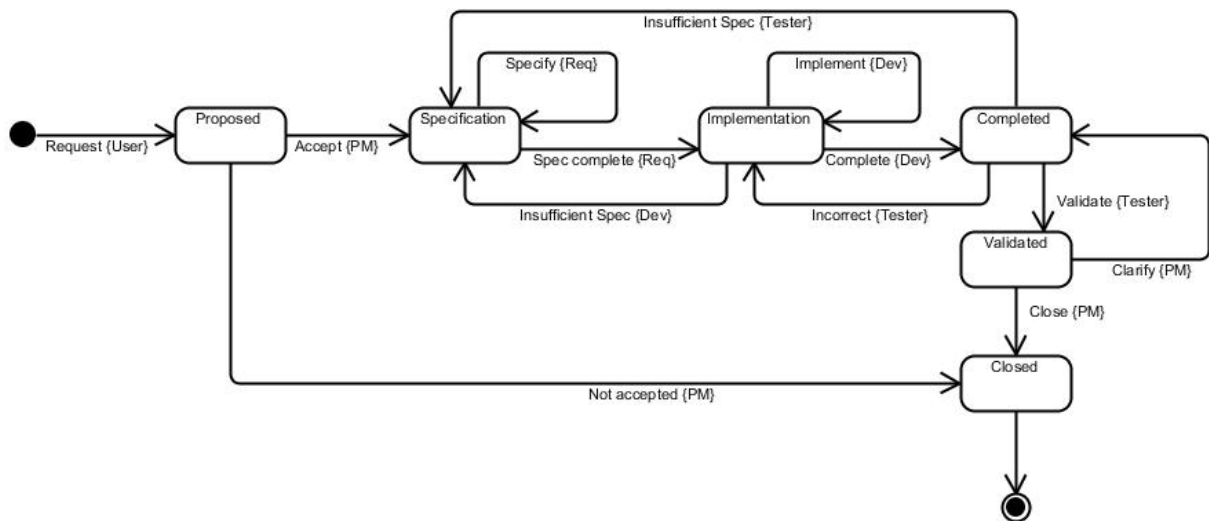
Figure 65: New feature/change request.
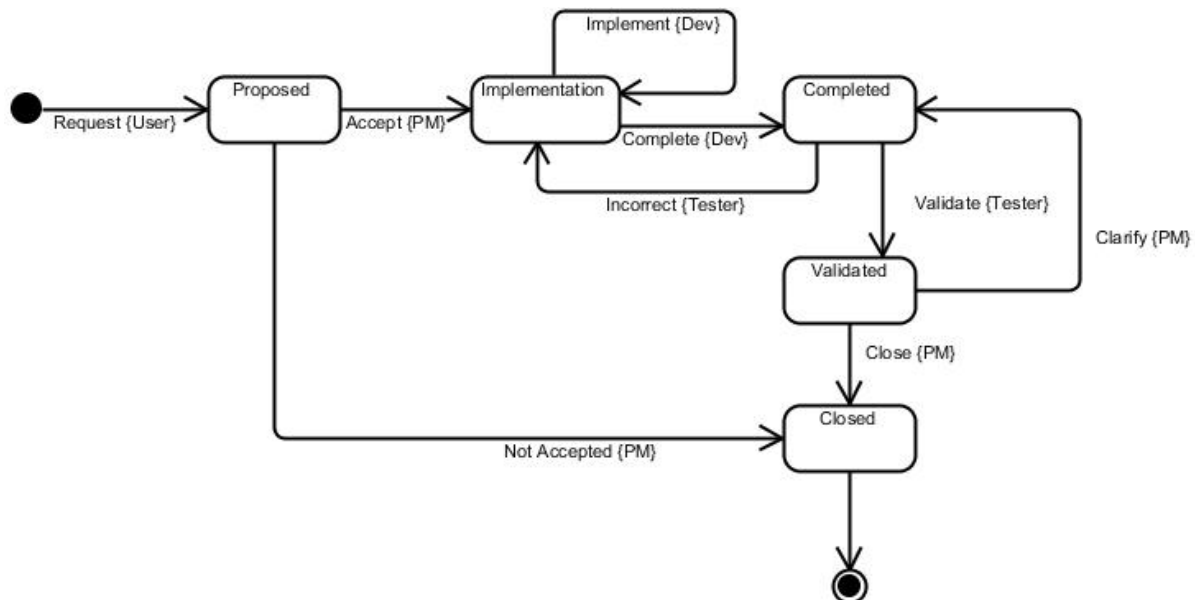
Figure 66: Improvement.

4.2.1.3 Lessons learned

Difficulties and Solutions

The definition of use-cases in the beginning tended to be useful to get a basic understanding of the tool chain. However, designing a modular platform lead to the need of many changes for these use-cases. The use-cases and the implementation needed to be reworked during the development and evaluation.

Sharing the development work among three different project partners made synchronization of the work difficult in the beginning. Especially the usage of three version management system, three bug-trackers and the exchange of files lead to conflicts. This was finally solved by defining clear workflow schemes and by unifying the development to on central version management and bug-tracking system.

Different scenarios for usage of the tools made a modular approach challenging as interfaces of the software components needed to be matched in all types of operation, e.g. no availability of database or server systems.

The .NET interaction with MATLAB has been proven to be a very powerful mechanism to add functionalities to interact with the server systems.

Future Recommendations

The modular approach of the software should be extended. Especially the interfaces could be more standardized and could offer additional features like a command line interface. As well the integration of additional external tools beyond MATLAB could be very useful.

4.2.2   Swedish VMC

The tool developed in the Swedish VMC is called FOTware and is based on a modular and customisable graphical user interface (GUI). The implementation described below refers to the FOTware version v4.0. FOTware development is based on previous experience in the Swedish VMC in field operational test data analysis from previous projects such as TSSFOT (Chalmers: SAFER: TSS Naturalistic Field Operational Test (FOT) - Phase 1, 2011) and SeMiFOT (Chalmers: SAFER: SeMiFOT, 2011). In euroFOT, FOTware is used for data quality assessment, hypothesis testing, outlier analysis, and video annotation.

4.2.2.1 Implementation

FOTware is entirely developed in MATLAB using an object-oriented programming approach. FOTware is designed to be a completely modular and customisable tool. This means that FOTware can be modified by the user and can offer endless number of different configurations (combinations of modules) to respond to the different analysis needs and to the different users' desires. Modules communicate with each other through events which can be listened to and generated by the modules.

A core module is responsible to 1) enable creation of new modules, 2) offer basic data replay functionalities, and 3) save and load FOTware configurations. The other modules enable the other functionalities which are coming from the functional requirements described in Chapter 3. Each module is presented in a new window and can be more or less complex depending on the specific functionalities which it implements. A possible configuration of FOTware is shown on Figure 67.

Figure 67: Screenshot from a two-screen possible configuration of FOTware.
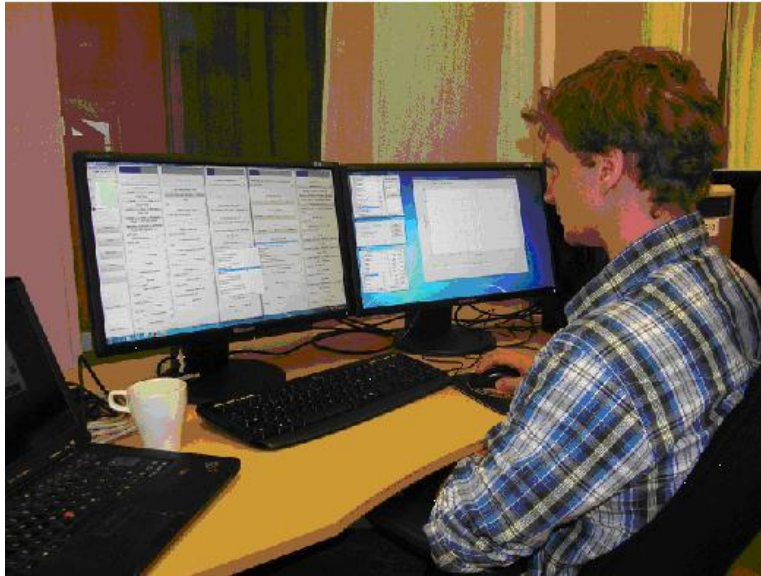
Figure 68: Annotation workstation with FOTware.

1 – Data player

The data player is the core module in FOTware and controls the playback of the data. Modules can be created manually from a list in the player or automatically via scripts. A slider bar displays the current position in time and can be dragged to jump to other positions. The sliding is instantaneous and all plots and video windows are continuously updated as the slider is moved. By using the mouse scroll wheel the user can change the playback speed if the data player is in *play* state. If the player is in *paused* state the scrolls will shift the current position in time. The data player displays information of the current trip date, segment and time. Closing the data player closes all other currently open windows and stores the setup. When FOTware is started again the user can choose to load the configuration with the same windows layout and settings as the previous one.
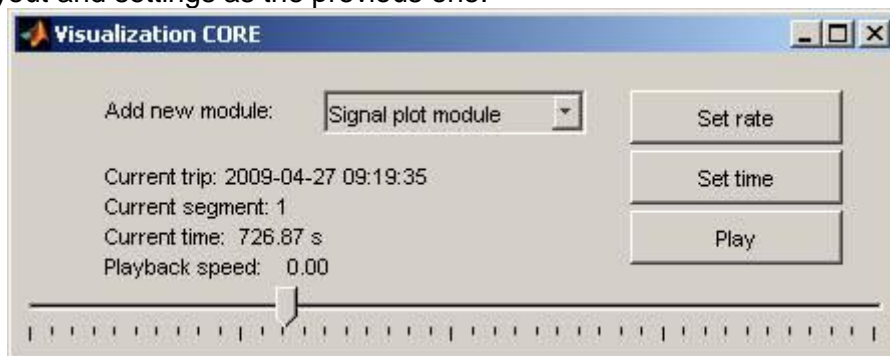
Figure 69: Data player.

2 - Video viewer

The video viewer displays video sequences by using ffdshow (a media decoder and encoder) (ffdshow tryouts 2011) and videoIO (a library providing access to video files using a wide variety of codecs in MATLAB) (SourceForge.net: videoIO Toolbox for Matlab 2011). The video playback is controlled by the core module.

The video source in the video viewer can be changed by choosing the desired camera view from a popup menu. It is also possible to choose among different camera views by pressing any numeric key from 1 to 5, where each of these numbers makes reference to the following available views, Figure 70:

1. Forward view

2. Driver view

3. Feet view

4. Rear view

5. Eye tracker view.

Figure 70: A sketch of a video display module when the forward camera view is chosen.

3 - SQL module

This module handles the access to the euroFOT Oracle database, the execution of SQL statements, and the retrieval and display of data. Figure 71 shows a sketch of this module in which the following main items can be identified:

- **Tree menu** that shows the hierarchy and names of the euroFOT tables and signals.

- **SQL-statement history menu** that keeps the statements that have been executed during a certain analysis session.

- **SQL-statement text field** that contains the statement that is executed when the button labelled "Execute" is pressed. Before executing any SQL statement or populating the DB tree menu (which requires reading the database metadata information), the user will be prompted for access credentials and should fill in the fields of the form shown in Figure 72. The options "Segmentise data" and "Expand segments" will complement the SQL command and change/adapt the way in which the data is retrieved.

- **Import and export buttons** that let the user transfer SQL statements to and from the simple SQL editor that is shown in Figure 73.

- **Information fields** that show relevant messages to the user such as the query execution status, the total memory used to retrieve the data, the time it took to retrieve the data, etc.

- **Data table** that shows the retrieved information organised in columns and limited by a predefined maximum number of rows.
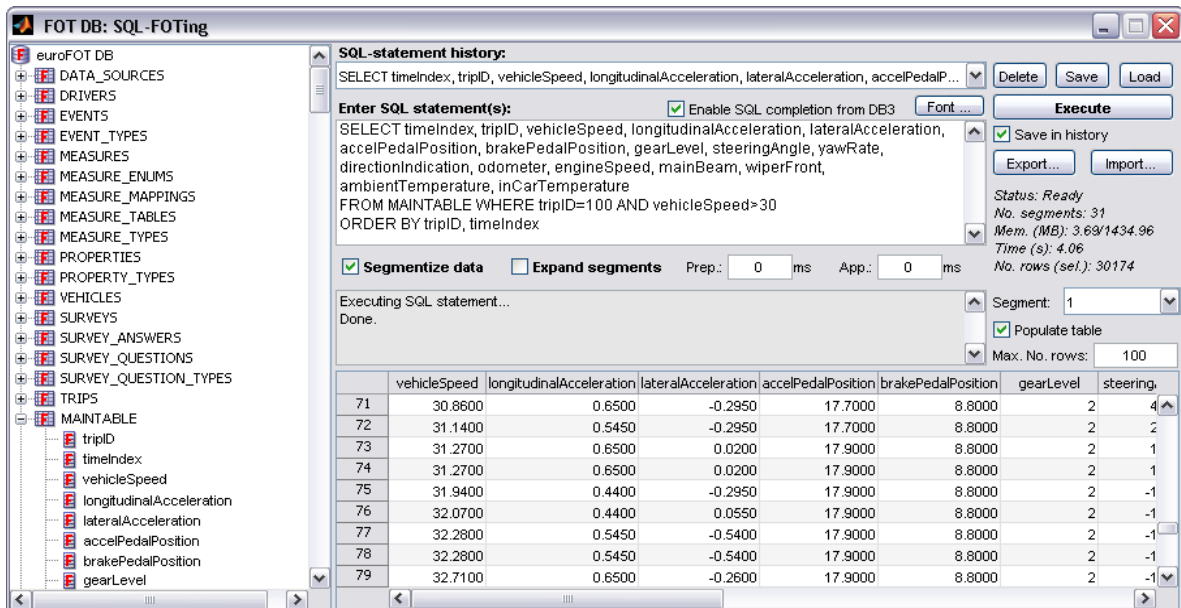
Figure 71: A depiction of the SQL module after successfully executed an SQL statement.

When the user opens the SQL module, he is prompted for DB credentials and needs to provide his/her username, password, the DB name and URL. These two last items will be configured when installing FOTware and will be available by default. However, the user is able to introduce different values in case that is required. It is also possible to keep the DB connection active and valid (and to provide credentials only once) during a complete MATLAB session by choosing to save the user password.
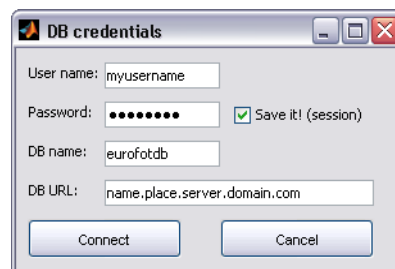
Figure 72: A sketch of the form the user should fill in with his/her database credentials.

With the SQL editor the users can modify existing SQL files and create new ones as well. Once the editing has ended, the user can transfer the text to the main SQL module just by pressing the button "Import".
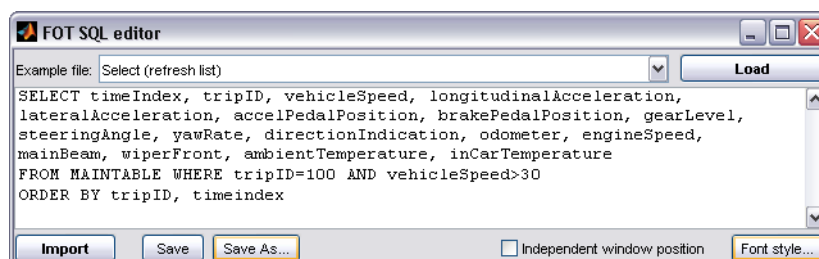
Figure 73: The SQL editor window.

4 – File browser

This module enables folder browsing and presents the user with a list of files which are possible to be loaded from the folder, Figure 74. The user opens a file by clicking on the file name. A check box is also available for the user to decide whether eye tracking data should also be read. The data currently loaded is displayed to the user. The user can also write directly (or copy and paste) a folder location to have access to it.
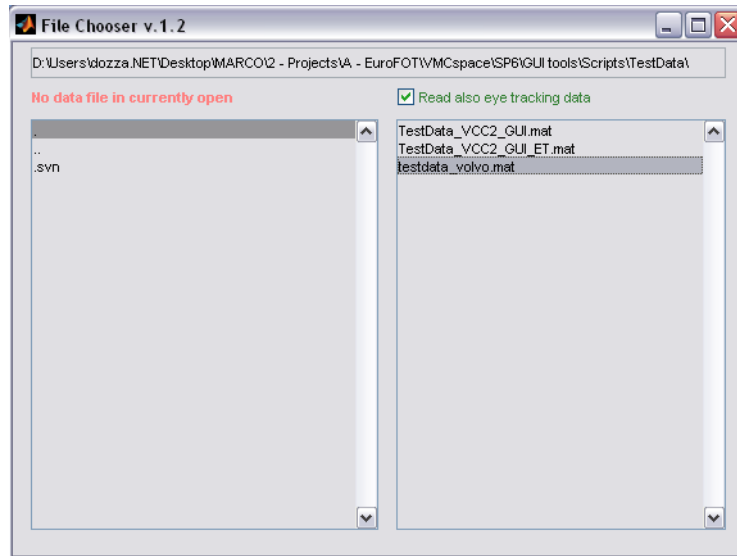
Figure 74: The file browser module.

5 - Signal browser

The signal browser displays the name and source of all the signals that are currently loaded and allows the user to select one or more signals that can be plotted (in a signal plotter module) or visualised (in a signal viewer module). Figure 75 shows a representation of this module when some data has been loaded. The elements of the signal browser are:

- **Signal table.** It lists down the signal names and sources, and allows the individual and multiple selections of signals.

- **Signal selection buttons.** These can select or unselect all signals present in the signal table.

- **Target module menu.** It lets the user select the target signal plot or signal visualisation module in which the selected signal should be plotted or listed down.

- **Plot type menu.** With this menu the user can specify the desired plot command among: linear plot (plot), histogram (hist), and cumulative distribution (cdf).

- **Signal preview checkbox.** When selected, the signal browser will plot a preview of the signal whose cell has been selected by the user, taking into account the plot command specified by the plot type menu.

After specifying appropriate values for all the elements of this module, the user can plot/visualise the selected signals by pressing the button "Done".
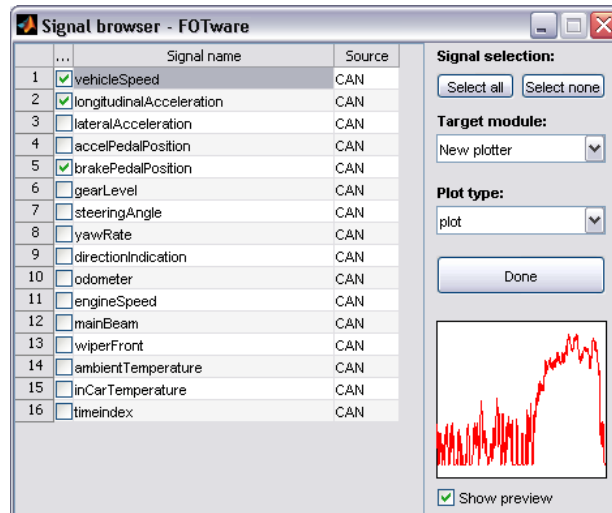
Figure 75: The signal browser showing the preview of the first signal of the table.

6 - Signal viewer

The signal viewer receives input from the signal browser and the event browser that specify the group of signals and events that should be plotted. It also presents a draggable doted red line that works as a time slider that affects all time-dependent modules (such as video viewers).

A single signal viewer can have several signals and events plotted at the same time. Figure 76 shows a signal viewer in which three signals have been plotted. It is possible to modify some properties of the signal graphs by changing certain values in the table located below the axes.
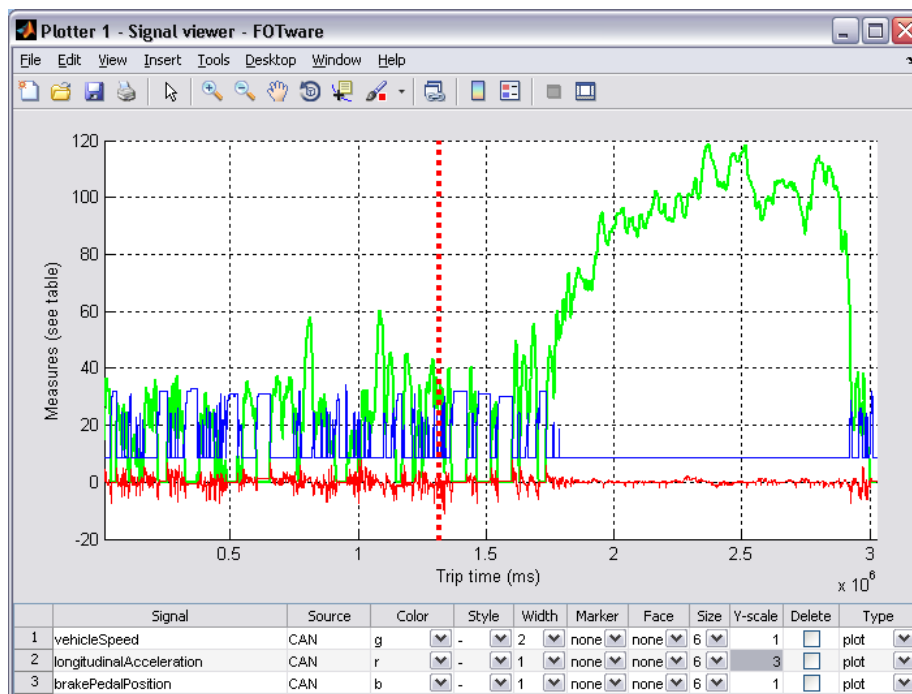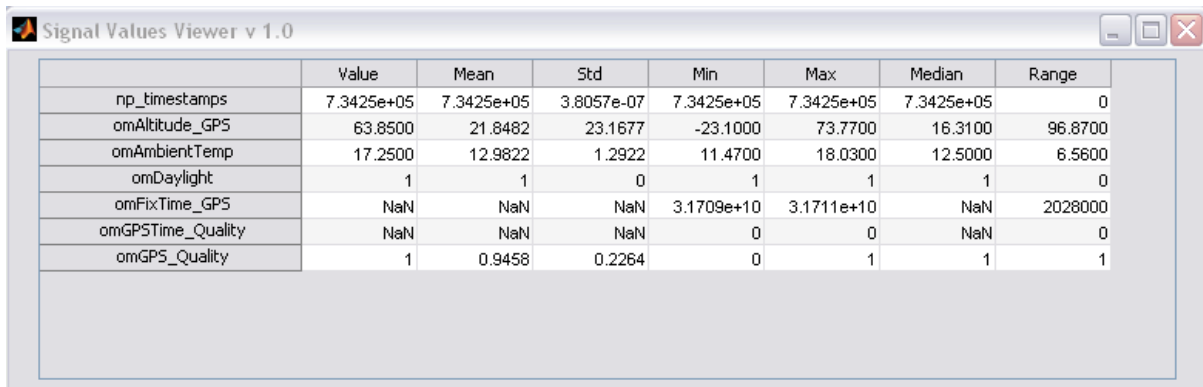
Figure 76: Three signals plotted in the signal plotter module.

7 - Data properties monitor

This module shows, using a table, the instantaneous values and statistical properties of currently selected signals from the signal browser, Figure 77.
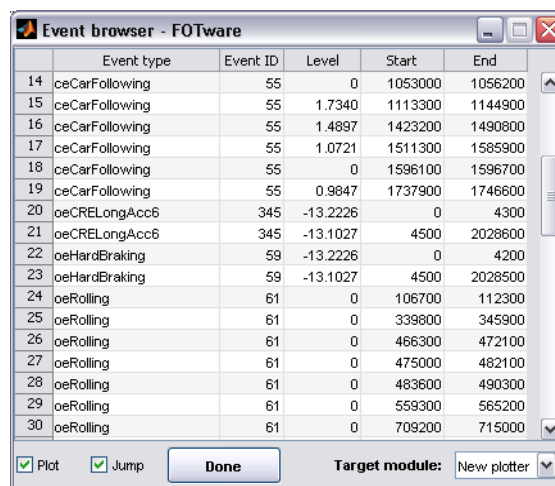
Figure 77: Statistical properties and instantaneous value from seven signals.

8 - Event browser

Events are points in time or time intervals that have a type name and numerical "level" value that has a type specific meaning. The event browser lists down all identified traffic events that are currently loaded. With this module it is also possible to plot events (as a Boolean or multi-level signal) and jump to the time in which they occur. The user can specify the target signal plotter window in which he/she desires to plot a certain event. Figure 78 presents a sketch of the event browser.

Figure 78: The event browser module displaying some identified traffic events.

9 - Video and event annotation

This module is intended to assist the user in video annotations. In order to launch this module, data must have been read (either with the file browser or the SQL module) and an event must have been selected with the event browser. Differently from any of the other modules – which have one and only one window for each instance – the number of windows in this module may vary. In fact, the number of windows depends on the number of annotations to be done.

This module is dynamically generated from an excel file. Depending on the content in the excel file the module can be used for different types of annotations and can have different appearances. The excel sheet must be generated by following specific rules in order to be interpreted by this annotation module. More specifically, there are 6 different types of annotations which can be described in the excel sheet (e.g. single choice, multiple choice, etc.). Furthermore, annotations can be organised into different sheets inside the same excel document. Each sheet generates a new window. A browser-window is also generated to enable the user to visualise, hide, or select the current annotation window by pressing the

buttons shown on the left window in Figure 79. Each annotation will be presented to the user as buttons, list box, or pop-down menu depending on the nature of the annotation and on the number of possible choices.

In the browser-window a table with the current status of each annotation is presented. By clicking on this table the user can plot the annotations in time. Further, the user can see the current status of the annotation work in a progress-bar at the top of each window. An overall progress-bar is also presented at the top of the browser-window. Finally, a check box in the browser-window enables the user to write the annotations.

When closing the annotation module the user is asked if the annotations need to be saved. If the user consents, annotations will be saved in a MATLAB file. This file contains all information needed to relate the annotations to the specific trip and event. The file has a unique name indicating the user, the time of annotation and the related trip (e.g. dozza_A_20_14045_11-Aug-2010_13_22_36.mat can be interpreted as user *dozza* made an annotation on event *20* in trip *14045* on the *11th of August 2010 at 13:22:36*).

Annotations' files are uploaded to the database by a *super-user* with rights to write in the database (typically not the annotator who is not likely to have such rights).
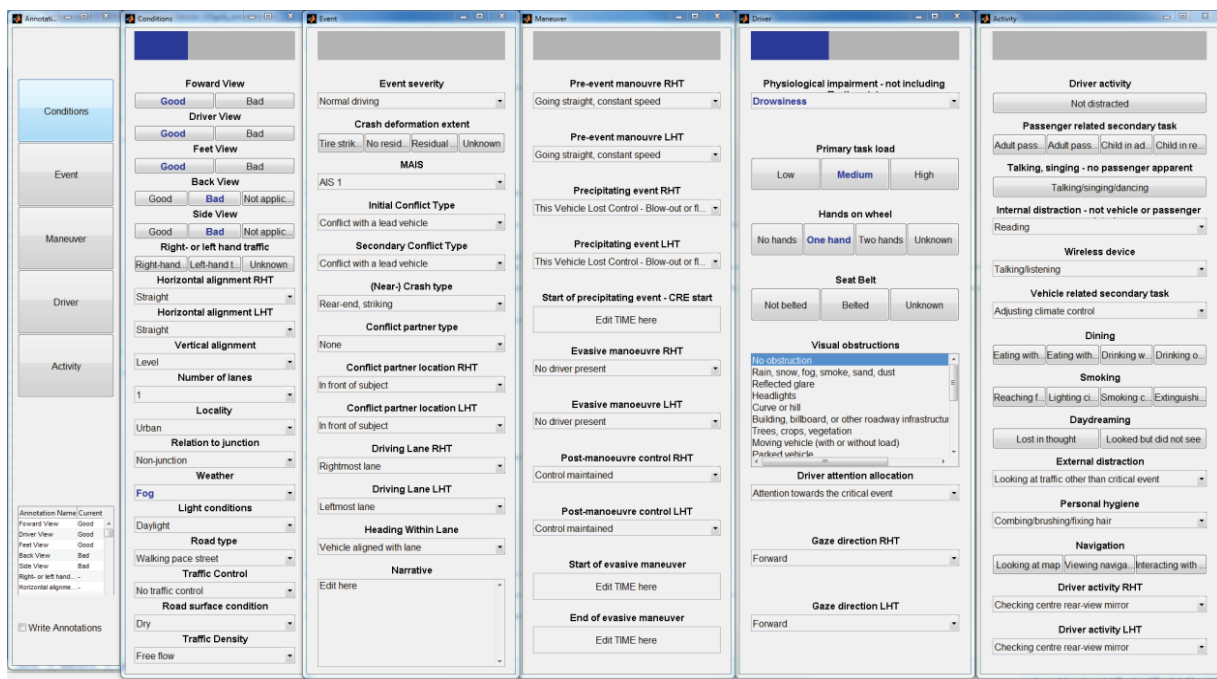
Figure 79: A possible configuration of the annotation module inspired by the 100 car study.

Unique and novel features of FOTware include modularity and customisability as well as the programming approach and the definition of events to communicate across modules.

FOTware implements the required functionalities listed in Chapter 3. Further, new analysis needs may raise when the user starts analysing and understanding the data.

4.2.2.2 Tests performed and quality assurance

FOTware releases and upgrades

Some details of the releases of different versions of FOTware can be summed up in the following list:

- Version 1.0: This release included the data player module, the file browser, signal viewers (with embedded signal browsers) and video player modules.

- Version 2.0: Besides fixing some bugs and optimising some routines of FOTware v1.0, the data property monitor modules for both CAN and eye-tracker signals were added in this version.

- Version 3.0: The updates in this release include an independent and more functional signal browser, an event browser, and video annotation modules.

- Version 4.0: This release includes an "SQL module" that handles the connection to an Oracle database and also manages the data retrieval procedures.

Management of improvement suggestions and bug fixing

All FOTware users were/are able to provide feedback to the tool responsible group (TRG) by using the short template report '*Template for reporting bugs and suggesting improvements for FOTware*' that is described in deliverable D3.2. All bugs that were found in each version were fixed for the release of a subsequent version and also as much suggestions for improvement as possible were taken into account.

Help and support

The help and support (H&S) service is provided by the TRG people and will be available during the extent of the euroFOT project.

FOTware users who know the TRG members can write emails and/or make phone calls to specific persons in order to get assistance. It is also possible to leave messages in a log book (where users can write errors, additional suggestions for improvement, etc.) that can be found in the FOT analysis room. However, this log book is recommended for non-urgent issues. All users that do not know who to contact should in first instance contact the analysis tool responsible at the Swedish VMC centre.

All trouble issues found and their constantly updated statuses shall be reported in one document. It is important to keep track of all relevant follow-up actions and also the name(s) of the assigned TRG member(s) who is (are) dealing with each support case. All TRG members should have access to this document in order to update it accordingly.

A FOTware user manual is also available. In this document users can find directions for making a proper use of this tool.

Since FOTware is modular and customisable, it is used by different users for different data analysis purposes. Further, since FOTware was developed in collaboration with SeMiFOT, also the SeMiFOT project will make use of it. At the current time the tool is used mainly for annotation of video and signal quality inspection. In addition, since the tool started to be used, there was a continuous and direct communication in between the users and the programmers to eliminate bugs and increase overall usability of the tool. At the present time the tool can only analyse one trip/event at a time, and can provide parameters to be used for statistical analysis, however FOTware enables only descriptive statistical analysis of individual signals from the CAN bus, or eye tracking. In the future, this tool will be also used for outlier analysis, and hypothesis testing. The extent to which new functionalities will need to be added for these new types of analysis is still uncertain at the present moment. In any case, statistical analysis is envisioned to be performed on commercial programs already developed for this purpose such as SPSS.

4.2.2.3 Lessons learned

The general experiences from FOTware are very positive and previous Swedish FOTs have shown the benefit of using a modular tool. It is also a large advantage to use MATLAB where the majority of analyses are carried out anyway. The scripting language provided through MATLAB allows several users to develop modules and features independently. Changes can automatically be incorporated through the use of version control system and no compilation or restart of the software is required.

Difficulties and Solutions

Initially there were concerns of the performance that could be reached in MATLAB, particularly for video playback. However, through the use of ffdshow a sufficiently fast solution could be achieved to play at least 6 video channels in parallel on a reasonably modern computer.

Although the FOTware is same between the partners in the Swedish VMC, there are still differences that depend on the location from where the FOTware is used. Thus configuration for a specific location was used which enables having 3 different setups, instead of changing the root paths for FOTware, location of video on the filesystem and the database settings.

Future Recommendations

For future development the modular approach is strongly recommended. MATLAB's COM object interface allows other application to control and interact with all scripts running in MATLAB, thus development can also take place outside of MATLAB.

4.2.3   German-2 OC

In German-2 OC, the data is stored as .mat files (MATLAB files) in file management system (no relational database). Visualisation of video data and some time-series data is done through a GUI built in MATLAB. For some deeper analysis, MATLAB scripts are written and executed on-demand.

4.2.3.1 Implementation

A MATLAB-GUI has been implemented that allows viewing the video together with selected variables of time series data. The function of the program is tailored to the data/folder structure used in the German-2 OC. Based on the naming and structure conventions, it selects automatically the video on the external hard drive that belongs to a selected trip. The program works with video data stored as single frames as well as with standard video format.

After starting the program, a drop down menu can be opened that shows all trips per driver (see Figure 80, trips for driverID 5 can be selected). Here, the experimenter can select the trip for which he wants to view the video.
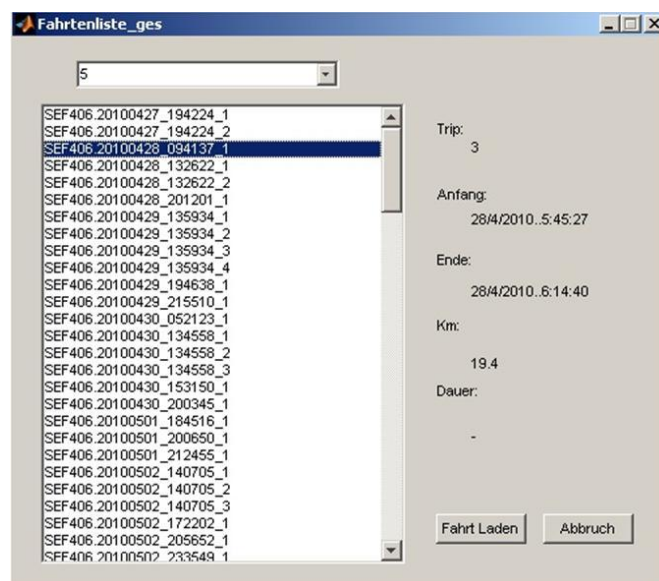
Figure 80: Selection of trips for driver ID 5.

After selecting a trip, the vehicle data and the video data are loaded. From all parameters in the mat-file (raw + map + derived variables), variables can be selected that shall be displayed in one of two diagrams, Figure 82, displayed on the right hand side of the video.
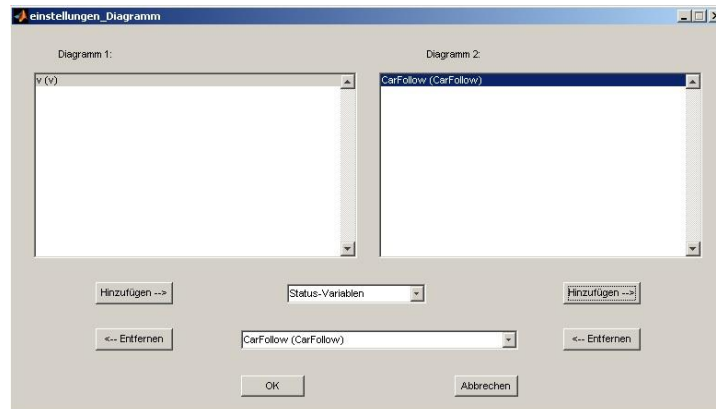
Figure 81: Selection of variables to be displayed together with video.

The GUI allows viewing the video while the selected variables are displayed over time. The length of the time window used for depicting the selected variables as well as the speed with which the video is displayed can be chosen.
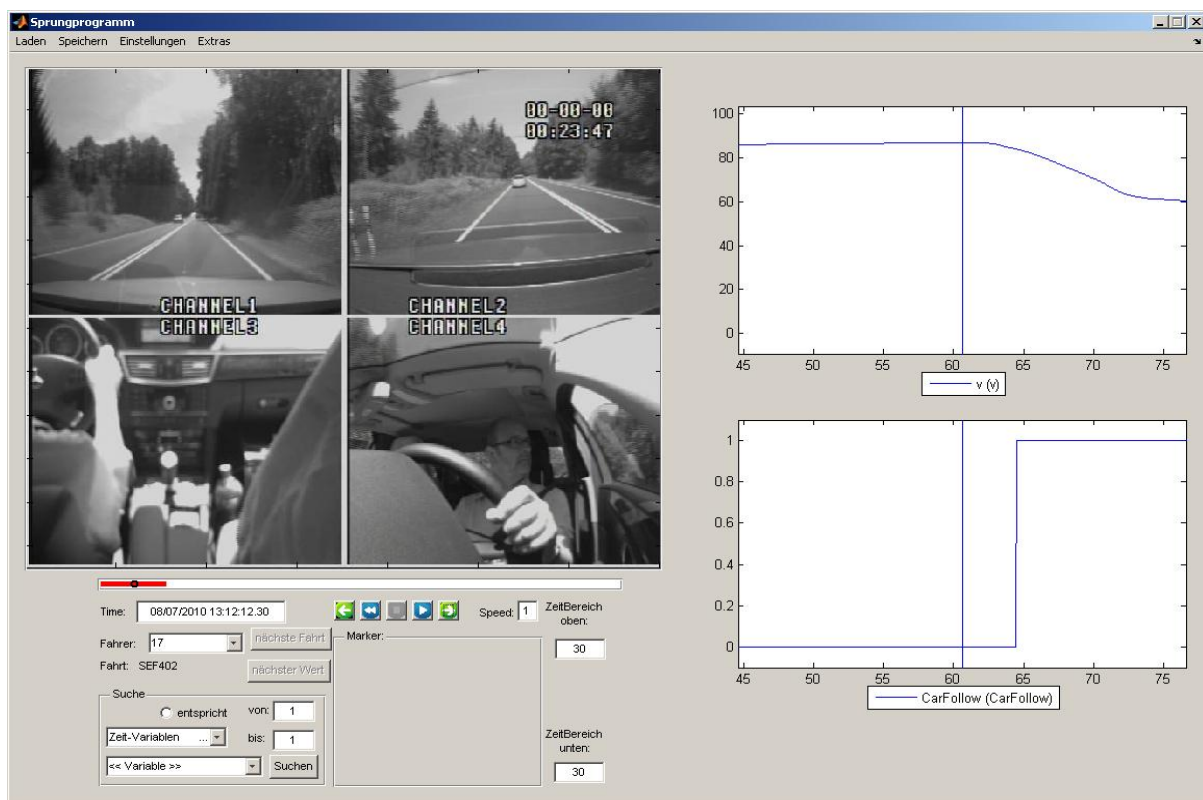
Figure 82: Display of video together with selected time series data.

Furthermore, it is possible to jump to defined events based on a selected variable, Figure 83. This enables the experimenter to systematically control for instance event detection algorithms based on the video.
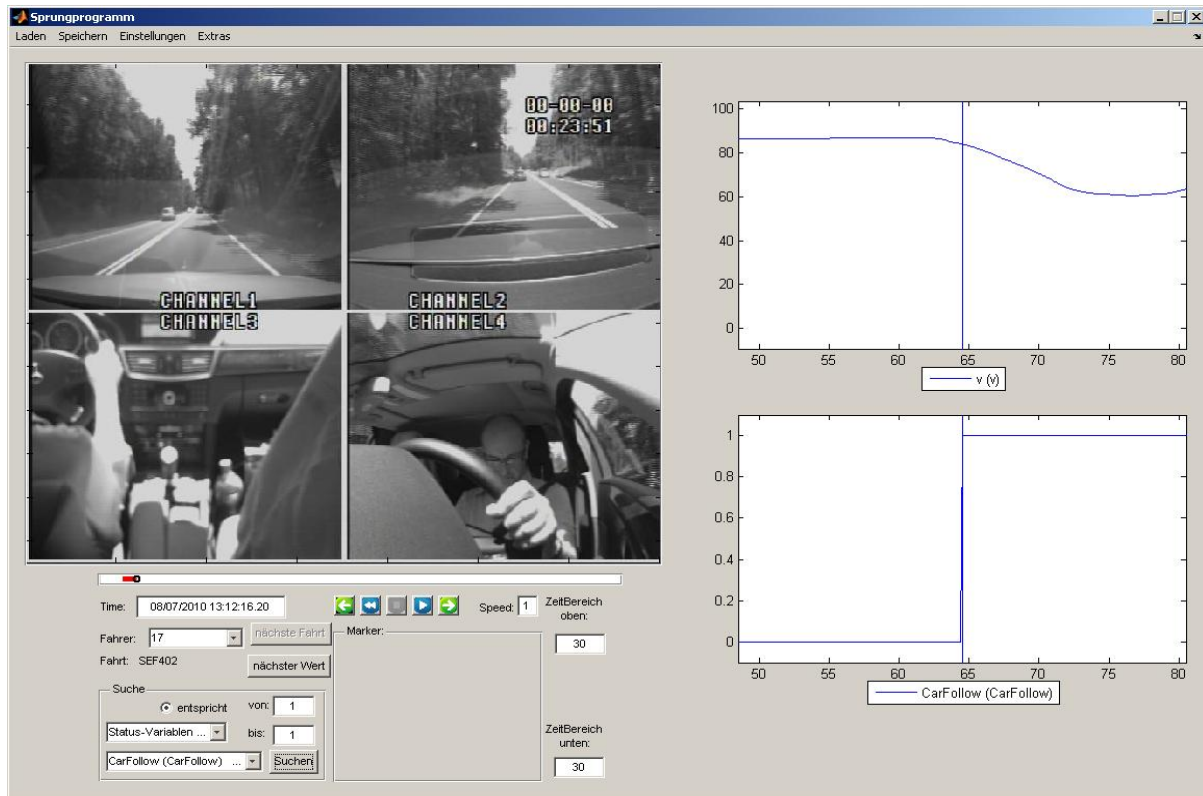
Figure 83: Jump to defined sections in the video based on values of a selected variable.

The calculation of parameters needed for the data analysis is not done based on a GUI; instead code is written in MATLAB to extract the needed information. Because the database consists of a large number of trips that are stored independently of each other, all indicators are calculated first per trip or per relevant event (e.g. incident). The resulting tables are the basis for the final data analysis. In the final analysis, the tables giving information per trip or event can be used to plot indicators and to calculate aggregated indicators per driver, per condition or overall.

The implemented routines create one table per driver. This allows a repeated extraction of the needed indicators already before all FOT-data is available. Whenever running a routine, it is checked for which drivers' data is available in all three conditions and which of those have no results table yet. The analysis is only performed on the data of those drivers that are new.

Routines have already been implemented that extract indicators for the following cases:

- Per trip (overview table)

- Per event

- Per input to navigation system

- Per situational category/combination of categories defined by WP6400.

4.2.3.2 Tests performed and quality assurance

The tool was tested by the tool responsible person at the pilot stage. The initial version was developed at BMW and then it was improved by the IZVW. There was very close interaction between users and developers, and the errors and bugs were corrected immediately, since there are few users. MATLAB scripts are written and executed on-demand.

4.2.3.3 Lessons learned

Difficulties and Solutions

Data handling (especially of video data) should be tested with a large amount of data already in the pilots: In the beginning, also the Daimler video data was stored as single frames instead of video format. Although there were a large number of separate files for each trip, the handling of the videos (e.g. copying) was no problem with the pilot data (about 50 hours / 3000km of data logging). After converting the first months of FOT data, it was realized that because of the huge number of single files, copying the data from Daimler conversion which uses ext3 (ext3 - Wikipedia, the free encyclopedia 2011) to IZVW server which uses NTFS (NTFS.com File System Structure, Recovery Software, Hard Disk Internals 2011) file format took longer than measurement time. As a consequence, the format had to be changed from single frames to video format and the database and data analysis tools had to be adapted. Especially the adaptation of the video tool took a lot of effort. Now the tool works with both, single frames and video format.

Future Recommendations

The whole chain of data pre-processing, upload and analysis should be tested (if possible) until PI-calculation before the FOT. Some problems with the signals (e.g. strange values) were realised very late in the whole process.

It has to be taken care that all partners who work with the raw data use the same software versions. This is important since it was found that many problems such as opening mat-files and playing video data were occurring because the versions of MATLAB used by the different partners differed.

4.2.4   Italian VMC

The Italian VMC does not use elaborate data analysis software like the other VMCs since this VMC only has questionnaires data. However, the Italian VMC uses standard tools such as LimeSurvey (LimeSurvey - the free & open source survey software tool! 2011), Microsoft Excel and SPSS (IBM SPSS software for predictive analytics 2011) for handling the data from the questionnaires and the information about the drivers.

After all the data from the questionnaires, both paper and electronically collected, are in LimeSurvey tool (see also Section 3.2.4), they are then exported into Microsoft Excel. In Microsoft Excel, the data are pre-processed and quality procedures are applied to prepare the data suitable for analysis according to the Data Analysis Plan of euroFOT-SP6.

The advanced statistical analysis will be done with statistical packages such as SPSS.

4.3 Remarks

This chapter described data analysis tools implemented in each VMC.

The data analysis software has accomplished the given goals and has met the user needs. Namely, the software is:

- User-friendly interface facilitating the annotation process;
- Flexible and not limited to specific research questions, allowing users to define annotation categories;
- Expandable so that the users could implement new modules and functions;
- Interactive with timeline-based visualisation of data;

- Efficient in retrieving information from both file management systems and relational databases.

Further improvement is possible by augmenting the tool with a command line interface and by integrating functions for statistical analysis.

5   Summary of Lessons Learned and Recommendations

In the previous three chapters, we have described the data management solutions used in euroFOT. The variety of solutions used allows us to learn a lot. Here, we summarise the lessons we learned from this project, from the data management point-of-view.

It is our hope that the lessons we list here and the recommendations we make here could benefit future projects.

The following way of work proved positive:

1. Using common templates and common terms to gather the requirements were very helpful, in particular these help during harmonisation and prioritisation of the requirements.

2. Working together for common interests has proven very constructive. For example, the French and Swedish VMCs worked together in defining requirements specification for several sensors, conducting market survey, as well as in evaluating some of the possible solutions. This increases the throughput and reduces the burden on both VMCs.

3. Collaboration with SP2 (the OEMs that know the "inside and outside" of the vehicles), SP4 (responsible for study design, performance indicators and list of measures), SP5 (that manage the FOT) and SP6 (responsible for analysis) from early time was absolutely useful and necessary. Their inputs are the base requirements for the data management.

Review on quality assurance:

4. Setting quality threshold is still a challenging issue in FOT. This issue has been raised by previous projects and FESTA, but it is worth mentioning again. Different quality threshold might be needed for different types of analysis (this applies to signals, measures, and video and/or other sensors requirements). One cannot really exhaustively list, in advance, all the possible analysis that could be done.

Other reflection:

5. Non project members often ask "why not use common solutions for all VMCs?" While using common solution(s) seems ideal, it was inevitable to finally choose different solutions as the requirements from the different VMCs (in fact from different partner organisation) are different. Often the solution has to fit partner's own standard, existing IT infrastructure and expertise. In the spirit of research, having different solutions is in fact very good, as it gives us an opportunity to assess the suitability of the different solutions for FOT.

5.1 Data Acquisition Systems

The following works well:

6. Having specialised/focused groups to enable joint efforts in performing certain tasks, for example investigating some specific sensors like eye tracker, radar, cameras, and map providers, were found useful.

The following could have been done better:

7. Unavailability of suitable DAS fulfilling euroFOT requirements in the market (particularly for a large scale FOT) has left us with no other solution than to partly develop our own solution (in particular, for the Swedish VMC). It would have been convenient to build a complete test rig for emulating the environment. For example, the Compact Flash problem would most likely have been detected in an earlier stage

if the system had been left continuously running for one or two weeks in a test rig. The pilot car was run for several months without failure.

Review on hardware, software, technical methods and tools used:

8. Data must always be saved in the DAS (either in the internal or external storage like hard disk or SD cards), because wireless connection is not always reliable. GPRS connection issue was experienced quite often in German-1 OC.

9. If the vehicles are driven across many different countries and GPRS is used for transferring data to the home country, it is important to understand the services from the different network providers. As this could affect the set up of the transfer mode from the DAS to the server (in this case, the server at the VMC centre).

10. Proper test and care should be taken when IR illumination is used as part of the DAS (for example, as part of eye tracker). Although most drivers did not get any problem, several drivers experienced dry eye but got used to this after a while.

11. Compact flash cards and SD cards are among the common DAS problems experienced in euroFOT. The consumer grade compact flash and SD cards have had to be replaced with those of industrial grade, not long after FOT started. Therefore, it is recommended to use compact flash and SD cards with industrial grade from the beginning.

12. The use of office (non-automotive) hard disks as storage medium in the DAS collecting video data was considered OK, especially when the DAS is equipped with hard disk health check functionality (as in the SAFER-euroFOT DAS case).

13. Inbuilt diagnostics in the DAS and web-based monitoring tool are necessary to be able to follow up the status of the DAS systems, to minimise risk of losing a lot of data due to undetected error(s), and for general quality check.

14. Failing connectors have been the main source of DAS errors in the Swedish VMC. In most cases, these are due to human errors when assembling and/or handling connectors. It is recommended that all personnel involved in installation must be well trained to do the job, and that proper quality check must be done after every installation.

15. It is recommended to give extra attention to cameras during installation and daily monitoring. Loose cameras were often found in the Swedish VMC.

Be ready with some unexpected work:

16. If cable harnesses and mounting brackets are to be manufactured internally, beware that one cable harness tool about 8 hours to manufacture.

17. Many parts or components used in DAS might not be available in automotive grade. It is recommended (particularly if building in-house solution) to give plenty of time for making DAS' components acceptable to OEM's standard.

Other reflections:

18. Packaging the main DAS unit with all of its components per vehicle-based was found helpful during installation. This is especially important when there are many loose things, for example, when DAS includes cameras and extra sensors.

19. Related to lesson learned #11, the PC (SAFER-euroFOT DAS) has to be opened to replace the CF card; this was a tedious job. Therefore, it is recommended to find a PC (if using) with the CF card (or any storage medium used) accessible from the outside.

5.2 Databases & Data Storage

To summarize the aspects related to database and storage we can see a number of lessons learned. Since the VMCs have different implementations and solutions the lessons learned varies.

Handling large amount of CAN and video data has been a challenge. Both copying and also converting video files is time consuming. This was solved by an adaptation in the database and analysis tools at German-2 OC. In the Swedish VMC, a cluster environment has been used to increase the speed which in turn brought problems handling unique row id's.

Another issue has been the upload into an SQL database. There have been difficulties with the design and performance problems. There are also different storage solutions and from these we can learn about the benefits and disadvantages of choosing a SQL-based or a file-based data storage.

The following works well:

20. Separating the handling of incoming disks/data from pre-processing stage has been a success. Especially since all steps in the pre-processing were not finished before data collection started.

The following could have been done better:

21. We spent a lot of time and effort on gathering requirements for database and data storage in the beginning of the project. It might have been more effective to use a more iterative process. Since in practice, some of the requirements were changed and the solution was adapted, however the requirements specification was not updated to include those changes.

Review on pre-processing, upload, database and data storage:

22. The whole chain of data pre-processing, upload and analysis should be tested (if possible) until PI-calculation before the FOT. Some problems with signals (e.g. strange values) were realized very late in the whole process.

23. The data pre-processing and upload procedures should be flexible and allow changes after the FOT has started. This is necessary because new problems might occur that did not exist in the pilot.

24. Evaluate at an early stage all the software needed. At the Swedish VMC some software packages (decoding CAN, NAVTEQ map data) needed Windows as operating system, causing a more complex automated flow on Linux-based high performance computing (HPC) environment.

25. Overall the cluster environment using MATLAB parallel toolbox has worked very well. The drawback has been that management of unique identities has become more complex.

26. Pre-processing and analysis of these large amounts of data requires 64 bit memory structures and software support.

27. A "sandbox" database including a subset of the data should be created. When developing and refining scripts it is much more time efficient to test on a smaller dataset. When satisfied the analysis can be executed on the whole dataset.

28. To be able to use SQL for analysis on billions of rows in a single table, effective index needs to be created. It is also vital to consider the storage needed for all indexes as indexes could take a lot of space.

29. Use and analyse "explain plans" before executing queries. By doing this, badly written SQL commands could be avoided.

30. Try to fully use the power of state-of-the-art servers utilizing many cores. By using parallel techniques in the database server or splitting the queries into smaller pieces can boost the performance.

31. Consider the cost for making backups of the database. If raw or pre processed data is stored and has backup there might not be need for taking backups of the continuous data tables. The time for re-uploading data might take longer time but it is important to state the service level agreement for downtime.

32. There is a great cost (in terms of computing time) for performing joins when dealing with large tables. By extensive de-normalization of the schema and even duplicating data analysis can be faster.

33. Making data access based on roles proved to be positive.

34. By converting video data from xvid to h264 the data storage need was reduced by 40% at Swedish VMC. The process is very CPU intensive and a cluster of computer nodes were needed to perform this task.

35. The large amounts of video data exceeded the filesystem limits of storage appliances. Therefore a server providing logic to be able to create a single mount point was needed. For future projects different filesystems and appliances could be evaluated; for instance by using ZFS or equivalent for almost unlimited shares.

Review on management of questionnaires data:

36. In order to reach more drivers and improve response rate, it is very good to offer different options for filling in questionnaires (i.e. paper based or web-based). However, if participants agree, it is recommended to use a web-based questionnaire; this will prevent data transcription mistakes.

Other reflections:

37. In contrast to FESTA's suggestion (FESTA-consortium, D2.2: Data requirements for FOT methodology 2008) to use relational database to store all non-video data, we thought it might be wise to be open to other available and emerging technologies like column-based databases, Hadoop database (HBase) (HBase - HBase Home 2011) and a data warehouse system for Hadoop (Hive) (Welcome to Hive! 2011) for managing large amounts of data for future projects of this scale or bigger.

38. In euroFOT, filesystem was also used in addition to the relational databases. For example, the Swedish VMC and German-2 OC stored the FOT data as Mat-files. This is to allow analysts to work through the database and/or directly with MATLAB. The effectiveness of using file management system (FMS) for complex analysis of large FOT data is yet to be tested. Among the drawbacks of using FMS compared to the database management system (DBMS) are: 1) Data access on FMS is not as flexible and granular as in DBMS; 2) consistency check on data is not at all enforced in filesystem.

39. Due to budget constraints, the use of special features, like compression and partitioning, of relational databases (for example, Oracle) and proper data warehousing has not been thoroughly explored in euroFOT. It would be interesting to see how these features could enhance the solutions we adopt.

5.3 Data Analysis Tools

The following works well:

40. A crucial step in the development phase was testing the analysis tools by as many users as possible. The feedback from the users was valuable input for finding the

errors, fixing the bugs and making the improvements according to the user's suggestions.

41. SQL software tools may require a fairly high level of knowledge to use, so it was advantageous to develop easy-to-use graphical user interface modules.

42. The development process of data analysis tool and improvements on the data format and structure was an iterative process. While using the analysis tools, the data formats that were not consistent with the design and defined structure were also found and corrected.

43. The modular approach helped using the same modules in different use cases and/or on different machines.

The following could have been done better:

44. The modular approach of the software should be extended. Especially the interfaces could be more standardised and could offer additional features like a command line interface. As well the integration of additional external tools beyond MATLAB could be very useful.

Review on technical methods and tools used:

45. The operating system had a relatively slow data transfer rate for networking operations. Given that the data viewing for the annotation was performed through a network link, it tended to be a lengthy process, which in turn makes the annotators waiting to load the video for few minutes.

46. Object oriented programming in MATLAB has a steep learning curve; even for the experienced programmer it took time to change the syntax from the usual structured programming. Object oriented programming in MATLAB might be too complex for small GUIs and debugging could be difficult since the errors are not shown in the corresponding line.

47. It is apparent from the design of the platforms that we consider video to be a very important source of information when dealing with naturalistic driving data. Our view is that, in most cases when analysing driver behaviour in naturalistic data, it is necessary to include analysis of the driver's actions and the context (in-vehicle as well as surrounding traffic environment) based on video. The video annotation module presented here has been significantly inspired by the 100-car naturalistic driving study, FESTA methodology and to optimize the procedures to fit different studies, several (parallel) projects have been giving input to its design. The different tools and interfaces needed in and between the different platforms will be further developed as new projects and data emerge, and the platforms will grow and develop to fit specific needs brought forth by researches and engineers.

6   Conclusions

This deliverable, D3.3, describes the achievements made by euroFOT partners (particularly in SP3) in terms of data management.

Five different data acquisition systems used in this project have been presented in this deliverable. First is the CTAG Datalogger II (CAN-only logger) that was used in German-1 OC and French VMC for the low level instrumentation. Second is the CEESAR videologging system which is a combined CTAG Datalogger II and video logger based on Nexcom VTC3300, with radar, eye tracker and lane tracker. Third is the SAFER-euroFOT DAS, an integrated CAN and video logger based on Nexcom VTC6100, with external accelerometer and eye tracker. Fourth is the BMW DAS, a CAN and video logger based on network attached storage, with two radar sensors. Fifth is the Daimler DAS which is a CAN, video and audio logger. All of these systems have been successfully used for the data collection phase in this project. In particular, the CTAG Datalogger II and the SAFER-euroFOT DAS were equipped in more than one hundred vehicles driven in several European countries for more than one year. Except the BMW DAS, all DAS are accompanied with a web-based monitoring tool to enable early detection of errors, as part of quality management strategy.

How the collected FOT data are transferred from the DAS to the VMC centre and then processed in the different VMCs are also presented here. German-1 and French VMC use full wireless transfer for the CAN-data. All VMCs that collect video employ manual data pick up and only use wireless to send status reports.

The pre-processing and uploading steps are discussed at length in this deliverable. The database and storage vary depending to the expected total size of data. German-1 OC, with an estimated 12TB of data, uses Oracle for their database engine. The French VMC stores their CAN data, estimated to be 2TB (total with video is estimated 8TB), in a MySQL database. German-2 OC stores their subjective data in MS Access, but maintains their objective data as MATLAB files. In the Swedish VMC, the CAN data are stored in an Oracle database and as MATLAB files in the file server, where the videos are also kept. Having only questionnaire data, the Italian VMC uses a MySQL database.

The base software for data analysis used in the different VMCs is detailed in this deliverable. While the look and feel of the software is different in the different VMCs, the functionalities are very similar. At the time of writing, the tools have been used to extract, visualise data from multiple sources, and do some simple analysis; in the Swedish VMC, the tool has also been used to annotate the video data for several months.

Furthermore, how we came to choose those different solutions as well as the general quality assurance throughout the whole data management chain adopted in this project are all summarised in this deliverable. The quality checks are done at many stages, for example before and after installation of DAS, during FOT operation through the web-monitoring tool, during pre-processing of the data, as well as when the data are uploaded. The data quality is measured per sample, per measure and per trip. As part of quality assurance, the installation strategies, software for installation verification, web-based DAS-vehicle monitoring tools, procedures of handling problems during data collection, procedures of data handling from vehicle to VMC centres are created and agreed with SP2 and SP5.

47 lessons learned from this project, from a data management point of view, are highlighted. Using common templates for gathering requirements and having specialised/focused groups to enable joint efforts in performing certain tasks have been found fruitful. Compact flash cards, SD cards, failing connectors and loose cameras are the common DAS-hardware issues. CF and SD cards with industrial grade are recommended. Making sure that all personnel involved in installation are well trained to do the job is a must. Furthermore, giving extra attention to the installation of connectors and cameras is recommended. On the data handling procedure, separating the handling of incoming disks/data from pre-processing stage has been a success. Considering the cost for making database backups, if the raw or

pre-processed data are stored, then there might not be a need for taking backups of the continuous data tables. As described above, in addition to storing non-video data in the standard relational database, some partners also stored these data as MATLAB files to allow analysis, both through database and/or directly using MATLAB. The use of special features (like partitioning and compression) of relational database (e.g., Oracle) has not been thoroughly investigated in this project. The effectiveness of the database and storage solutions we chose is yet to be tested to the limit; at the time of writing this report, not all of the total collected data at each VMC has been uploaded to the respective VMC's database and storage. However, with so many other technologies available, for example column-oriented databases, Hadoop database (HBase), and the data warehouse for Hadoop (Hive), it is wise to be open to the potential benefits of those technologies or similar for the management and analysis of FOT data of the scale of this project or bigger. Regarding the data analysis tools, the modular approach adopted here has allowed reuse of the same modules in different use cases and/or different machines. It is felt that the modular approach could still be extended, the interfaces of the tool could be made more standardised, additional features like a command line interface could be added and external tools beyond MATLAB could be integrated.

In summary, the best suited data acquisition systems, the required infrastructure and services for data storage, the base software for analysis, general quality assurance as well as support in installation and maintenance strategies to both SP2 and SP5 have been provided to the project; SP3 has achieved all of its objectives.

Acknowledgements

We acknowledge and thank the following for their contribution to the work described in this deliverable:

Helena Gellerman	Chalmers
Riku Kotiranta	Chalmers
Dag Börjesson	Previously at Chalmers
Lena Kanstrup	Previously at Volvo
Jorge Leon Cano	Previously at VCC
Vassilis Kaffes	ICCS
Katia Pagle	ICCS
Leandro Guidotti	POLI
Roberto Tadei	POLI
Friedemann Kuhn	DAG
Mariana Just	BMW

We also would like to thank all other individuals that are not named here but have in some ways contributed to the work reported in this deliverable.

References

Chalmers: SAFER. (2011). Retrieved 08 08, 2011, from https://www.chalmers.se/safer/EN/

Chalmers: SAFER: SeMiFOT. (2011). Retrieved 08 08, 2011, from https://www.chalmers.se/safer/EN/projects/traffic-safety-analysis/semifot

Chalmers: SAFER: TSS Naturalistic Field Operational Test (FOT) - Phase 1. (2011). Retrieved 08 08, 2011, from http://www.chalmers.se/safer/EN/projects/traffic-safety-analysis/tss-naturalistic-field

Csepinszky, A., & Hagleitner, W. (2011). *Deliverable D5.2: List of drivers, vehicles, equipment, legal aspects and report on installation of the logging devices (Version 2.1).* euroFOT.

Dozza, M., Kessler, C., Petersson, M., Rakic, M., Tattegrain-Veste, H., & Heinig, K. (2010). *Deliverable D2.1-Updated: Specifications & Requirements for Testing In-vehicle Systems for Driving Support (Version 1.7).* euroFOT.

Embedded AntiSleep. (2011). Retrieved 08 30, 2011, from http://www.smarteye.se/productsoem/embedded-antisleep

euroFOT-consortium. (2008). *Annex I: Description of Work of European Large-Scale Field Operational Tests on In-Vehicle Systems.* euroFOT.

ext3 - Wikipedia, the free encyclopedia. (2011). Retrieved 08 08, 2011, from http://en.wikipedia.org/wiki/Ext3

FESTA-consortium. (2008). *Deliverable D2.2: Data requirements for FOT methodology.* FESTA.

FESTA-consortium. (2008). *FESTA Handbook Version 2.* FESTA.

ffdshow tryouts. (2011). Retrieved 05 30, 2011, from http://ffdshow-tryout.sourceforge.net/

Flament, M., Cavallo, F., Csepinszky, A., & Hagleitner, W. (2009). *Deliverable D5.1: Description of the VMCs and Common Guidelines (Version 1.1).* euroFOT.

Glossary - FOT-Net WIKI. (2009, August). Retrieved August 20, 2011, from http://wiki.fot-net.eu/index.php?title=FOT_Glossary

H.264/MPEG-4 AVC - Wikipedia, the free encyclopedia. (2011). Retrieved 08 08, 2011, from http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC

Hagedorn, J., Hailpern, J., & Karahalios, K. G. (2008). VCode and VData: Illustrating a new Framework for Supporting the Video Annotation Workflow. *Extended Abstract of AVI 2008* .

Hard Disk Sentinel - HDD health and temperature monitoring. (2011). Retrieved 08 30, 2011, from http://www.hdsentinel.com/

HBase - HBase Home. (2011). Retrieved 08 08, 2011, from http://hbase.apache.org/

Home:: Bugzilla:: bugzilla.org. (2011). Retrieved 08 30, 2011, from http://www.bugzilla.org/

http://www.ipcmax.com/datasheet/Nexcom_VTC_3300_Transport-PC.pdf. (2007, 08 03). Retrieved 08 08, 2011, from http://www.ipcmax.com/datasheet/Nexcom_VTC_3300_Transport-PC.pdf

IBM SPSS software for predictive analytics. (2011). Retrieved 05 30, 2011, from http://www.spss.com/

Jamson, S., Chorlton, K., Gelau, C., Schindhelm, R., Johansson, E., Karlsson, A.-S., et al. (2009). *Deliverable D4.2: Experimental Procedures (Version 0.7).* euroFOT.

Kircher, K., Heinig, I., & Brouwer, R. (2009). *Deliverable D4.1: Performance Indicators, Events and Situational Variables within euroFOT (Version 1.3).* euroFOT.

Kovaceva, J., Leon-Cano, J. A., Sanchez, D., Dozza, M., Selpi, Val, C., et al. (2011). *Deliverable D3.2: Base Functionality Data Analysis Software - Updated Version (Version 2.0).* euroFOT.

Kvaser USBcan Professional. (2011). Retrieved 08 30, 2011, from http://www.kvaser.com/index.php?option=com_php&Itemid=258&eaninput=7330130003576&lang=en&product=Kvaser%20USBcan%20Professional

LimeSurvey - the free & open source survey software tool! (2011). Retrieved 05 30, 2011, from http://www.limesurvey.org/

Matroska Media Container - Homepage. (2011). Retrieved 08 08, 2011, from http://matroska.org/

MOBILEYE AWS-2000. (2008). Retrieved 08 30, 2011, from http://www.clifford.nl/pdf/mobileye/UK-U-AWS-2000.pdf

MPEG4000XLP. (2011). Retrieved 08 30, 2011, from http://www.ampltd.com/dload/mpeg4000xlp.pdf

MySQL:: The world's most popular open source database. (2011). Retrieved 08 08, 2011, from http://www.mysql.com/

NTFS.com File System Structure, Recovery Software, Hard Disk Internals. (2011). Retrieved 08 08, 2011, from http://www.ntfs.com/

Oracle | Hardware and Software, Engineered to Work Together. (2011). Retrieved 08 08, 2011, from http://www.oracle.com/index.html

Parallel Computing Toolbox - MATLAB. (2011). Retrieved 08 08, 2011, from http://www.mathworks.com/products/parallel-computing/

Pea, R., Mills, M., Rosen, J., Dauber, K., Effelsberg, W., & Hoffert, E. (2004). The Diver Project: Interactive Digital Video Repurposing. *IEEE MultiMedia Volume 11, Issue 1* , 54-61.

Saint Pierre, G., Kessler, C., & Malta, L. (2011). *Deliverable D2.2: Report of the results of adaptation, in-vehicle implementations and piloting (Version 3.2).* euroFOT.

Selpi, Kanstrup, L., Val, C., Benmimoun, M., & Borgen, S. (2009). *Deliverable D3.1: Selection Procedure for Data Acquisition, Sensors and Storage for euroFOT (Version 1.1).* euroFOT.

SourceForge.net: videoIO Toolbox for Matlab. (2011). Retrieved 05 30, 2011, from http://videoio.sourceforge.net/

Statistical Analysis Software, SAS/STAT. (2011). Retrieved 05 30, 2011, from http://www.sas.com/technologies/analytics/statistics/stat/index.html

tortoisesvn.tigris.org. (2011). Retrieved 05 30, 2011, from http://tortoisesvn.tigris.org/

TrueCrypt. (2011). Retrieved 08 30, 2011, from http://www.truecrypt.org/

VTC 3300. (2007). Retrieved 08 30, 2011, from http://www.ipcmax.com/datasheet/Nexcom_VTC_3300_Transport-PC.pdf

VTC6100. (2010, 04 05). Retrieved 08 08, 2011, from http://www.rm-systems.co.uk/wp-content/uploads/2010/07/VTC6100.pdf

Welcome to Apache Hadoop! (2011). Retrieved 08 08, 2011, from http://hadoop.apache.org/

Welcome to Hive! (2011). Retrieved 08 08, 2011, from http://hive.apache.org/

Wittenburg, P., Brugman, H., Russel, A., Klassmann, A., & Sloetjes, H. (2006). Elan: a professional framework for multimodality research. *Proceedings of Language Resources and Evaluation Conference (LREC 2006).*

Xvid.org: Home of the Xvid Codec. (2011). Retrieved 08 08, 2011, from http://www.xvid.org/

ZFS - Wikipedia, the free encyclopedia. (2011). Retrieved 08 08, 2011, from http://en.wikipedia.org/wiki/ZFS

Zimmerman, P., Bolhuis, J., Willemsen, A., Meyer, E., & Noldus, L. (2009). The Observer XT: A tool for the integration of and synchronization of multimodal signals. *Behavioural Research Methods, Volume 41, Issue 3* , 731-735.

Annex 1 Keywords

Analyse

Annotation

euroFOT

CAN

DAS

Data acquisition system

Database

Data logger

Data storage

Data Analysis Tool

GUI

Logger

Market survey

Measures

Requirements specification

Sensors

Software Architecture

SQL

Upload strategies

Video

Video logger

Visualisation

Annex 2   Glossary

FOT related terms and definitions are taken from the official euroFOT glossary (Glossary - FOT-Net WIKI, 2009).

FOT (Field Operational Test)	A study undertaken to evaluate a function, or functions, under normal operating conditions in environments typically encountered by the host vehicle(s) using quasi-experimental methods.
Function	Implementation of a set of rules to achieve a specified goal.
Use case	A specific **event** in which a **system** is expected to behave according to a specified **function.**
Situation	One specific level or a combination of more specific levels of **situational variables**.
Situational variable	An aspect of the surroundings made up of distinguishable levels. At any point in time at least one of these levels must be valid.
Hypothesis	A specific statement linking a cause to an effect and based on a mechanism linking the two. It is applied to one or more **functions** and can be tested with statistical means by analyzing specific **performance indicators** in specific **scenarios**. A hypothesis is expected to predict the direction of the expected change.
Baseline period/phase	The part of the data collection during which the **function(s)** operate in ``silent mode´´, that is, they collect data, but do not give any signals to the driver. From the viewpoint of the driver the **function(s)** is/are off.
Treatment period/phase	The part of the data collection during which the **function(s)** are switched on by the experimental leader, such that they are either active all the time, or can be switched on or off by the driver.
Controlled factors	These are those factors that are kept constant within one **analysis.** The data are filtered such that only occurrences in which the controlled factors assume the intended values are selected.
Variable factors	These are covariates, they are not kept constant within one analysis, but their values are logged and their influence on the results is considered.
Performance indicator	Quantitative or qualitative indicator, derived from one or several **measures,** agreed on beforehand, expressed as a percentage, index, rate or other value, which is monitored at regular or irregular intervals and can be compared to one or more criteria.

DAS related terms

ADR	The European Agreement concerning the International Carriage of Dangerous Goods by Road.
Avi	Multimedia container format and filetype. Avi can be used as a placeholder for different media codecs as xvid or h.264.
CAN	Controller Area Network. A network standard for communication between control units in a vehicle.
DAS	Data Acquisition System. The in-vehicle hardware and software that constitutes a complete logging system.
Driving cycle	The period from the ignition is turned on until the ignition is turned off. It this document the word **trip** is also used.
EMC	Electromagnetic Compatibility.
Eye and head tracker	Hardware and/or software for tracking the movements of the drivers head and eyes.
HDD	Hard Disc Drive.
GPS	Global Positioning System
GPRS	General Packet Radio Service. A packet oriented mobile data service.
J1708	A network standard for communication between control units in a vehicle.
LIN	Local Interconnect Network. A network standard for communication between control units in a vehicle. Lower communication speed than CAN.
Radar	An object detection system that uses electromagnetic waves to identify the range, altitude, direction or speed of both moving and fixed object.
RTC	Real Time Clock. In this document it denotes the time and calendar circuit in a PC or similar computer.
Signal source	A source of signals, e.g. the CAN bus, a video camera or an eye tracker.
VCI	Vehicle Communication Interface. A piece of hardware for accessing CAN, LIN, and J1708 vehicle networks.
SSL	Secure Sockets Layer. A protocol that provides security and data integrity for communications over networks.

Database, storage, and analysis tool related

File Storage	The file storage area on the CMS, holding any data not stored in the database.
GUI	Graphical User Interface.
H.264	Video codec used to compress video
MySQL	An open source relational database management system
Oracle	A commercial object-relational database management system
Rsync	A software application for Unix systems which synchronises files and directories from one location to another while minimizing data transfer using delta encoding when appropriate.
SCP	Secure Copy Protocol, a secure version of the Unix remote copy (rcp) command. See SSH.
SDE	Standard Development Environment.
SQL	Structured Query Language, a database computer language designed for managing data in relational database management systems (RDBMS).
SSH	Secure SHell, a security protocol for logging into a remote server. SSH provides an encrypted session for transferring files and executing server programs. Also serving as a secure client/server connection for applications such as database access and e-mail, SSH supports a variety of authentication methods.
TRG	Tool Responsible Group
Use Case	A use case is a narrative description of the domain process. It describes story or a case of how an external entity (actor) initiates events and how the system responds to them. Thus it specifies an interaction between an actor and the system, describing the sequence of transactions they undertake to achieve system functionality.
VPN	Virtual Private Network, a private network that is configured within a public network (a carrier's network or the Internet) in order to take advantage of the economies of scale and management facilities of large networks.
XML	EXtensible Markup Language, an open standard for describing data from the W3C (World Wide Web Consortium). It is used for defining data elements on a Web page and business-to-business documents. XML uses a similar tag structure as HTML; however, whereas HTML defines how elements are displayed, XML defines what those elements contain.
Xvid	An open source video codec used to compress video.