Inf Syst Front (2013) 15:725–745
DOI 10.1007/s10796-013-9416-y

Towards a reference architecture for fuel-based carbon management systems in the logistics industry

M. E. Iacob · M. J. van Sinderen · M. Steenwijk · P. Verkroost

Published online: 13 March 2013

Abstract The current practice in the logistics industry is to calculate the carbon footprint of transportation activities based on the distance covered, using long-term fuel consumption averages per kilometer. However, fuel consumption may actually vary over time, because of differences in road characteristics, traffic situations, driving behavior, etc. Therefore, distance-based emission calculations are not accurate. Our approach is fuel-based and it calculates transport greenhouse gas emissions by obtaining the actual fuel consumption during trips via board computers installed in vehicles. Thus, we propose an architecture for a fuel-based Logistics Carbon Management System (LCMS) that monitors and collects real-time data about the fuel consumption during trips, and, consequently, calculates detailed and accurate carbon footprints of transportation services. Furthermore, this system is integrated with the logistics service provider's business processes and with typical software applications (e.g., Transport Management Systems and Board Computers). We validate and implement the proposed architecture by means of a prototype.

Keywords Carbon footprinting · Logistics carbon management system · Architecture · Logistics industry

M. E. Iacob (✉) · M. J. van Sinderen
Centre for Telematics and Information Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: m.e.iacob@utwente.nl

M. J. van Sinderen
e-mail: m.j.vansinderen@utwente.nl

M. Steenwijk
University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: m.steenwijk@alumnus.utwente.nl

P. Verkroost
Cape Groep, Transportcentrum 14, 7547 RW Enschede, The Netherlands
e-mail: p.verkroost@capegroep.nl

1 Introduction

The Intergovernmental Panel on Climate Change states that the recent warming of the Earth is very likely caused by an increased concentration of greenhouse gases (GHG)[1] produced by human activity. The European Union is committed to reduce GHG emissions with 20% by 2020, compared to 1990 (European Commission 2008). Since instruments, such as financial penalties and bonuses, will be used to control GHG emission in the economic sectors (IPCC 2007), companies are forced to gain insight in their GHG emissions. This is done through the calculation of the *carbon footprint* (Wiedmann and Minx 2008), which is a measure of the total amount of GHG or CO_2 emission caused by a product or activity.

Transport is the second largest contributor to carbon emission (after the energy sector). In the EU its combined GHG emission share is 19% and, as opposed to all other sectors, transport showed an increase of emission between 1990 and 2006. Globally, transport has a 23% share when counting CO_2 emission only (IEA 2009). Of all transport modalities, road transport is the largest contributor to carbon emission, as it accounts for 65% of all CO_2 emission caused by transport (Chapman 2007). Therefore, the logistics sector is likely to face legislation on CO_2 emission very soon. This puts a lot of pressure on Logistic Service Providers (LSPs) and increases the urgency for them to adopt a carbon footprint calculation system. Such systems provide insight in the GHG emissions, and may reveal opportunities for GHG reduction as well. Considering that the dominant driver of carbon emissions in logistics is fuel combustion, CO_2 emission reduction also promotes fuel and financial savings, which is another incentive for adoption.

[1] This study contains numerous acronyms. To improve the readability of the manuscript we have included in Appendix B a table of all the acronyms for quick reference.

Numerous protocols from various organizations have emerged over the last years, and calculation methods of carbon footprints are still under development. The current industry practice is to calculate transport emissions based on distance travelled, using long-term fuel consumption averages per kilometre. However, fuel consumption may actually vary over time, because of differences in road characteristics, traffic situations, driving behaviour, etc., which raises questions about the accuracy of distance-based emission calculations. Furthermore, transport services may stretch over multiple, sometimes multimodal, legs (i.e., legs for which different modalities, such as, road, train, air or water, are used to move the freight). While crossing warehouses, multiple shipments are often consolidated in one freight unit. This adds complexity to the concept of transportation carbon footprint calculation. Several generic Carbon Management Systems (CMS) that support such calculations have recently emerged on the market. However, the current state-of-the-art of these applications goes little further than rough estimates of a company's corporate carbon footprint. They do not take into account the specific characteristics of the transportation processes, as they are not designed specifically for the logistics industry. All the above reveal immaturity and significant limitations of the current methodological and technological carbon management state-of-the-art, in particular in the logistics industry. Therefore, our research goal in this study is *to design a reference architecture for a Logistic Carbon Management System (LCMS) that supports the fuel-based and real-time calculation of carbon footprints.* Thus, our approach addresses the afore-mentioned limitations, and is aimed at increasing the accuracy of carbon footprinting by using measurements of transport emissions based on the actual fuel consumption during trips provided by board computers installed in vehicles. The proposed architecture takes into account the business processes and information system (IS) landscape of typical logistics service providers. More specifically, by integrating systems, such as Transport Management Systems (TMS) and board computers (BCs) with a LCMS, the real-time monitoring of fuel consumption during trips, and the detailed and accurate calculation of product carbon footprints of transport services become possible. An implementation of the reference architecture by means of a prototype is also included in this paper. The prototype calculates emissions by linking incoming board computer XML messages indicating the actual fuel consumption with trip planning data from a TMS. A test case validates the correct behaviour of the prototype, and proves the feasibility of real-time fuel-based carbon footprinting.

Before concluding this section, we discuss the research methodology used in this study. The dominant methodological paradigms that are used to produce and publish information systems research are descriptive and explanatory approaches, borrowed from social and natural science (Peffers et al. 2008). However, these methodologies are less suitable for research in the area of information systems design. To compensate this shortcoming, Hevner et al. (2004) introduced the new paradigm of design science research, which *"creates and evaluates IT artifacts intended to solve identified organizational problems".* A series of very successful studies (e.g., Walls et al. 1992; Hevner et al. 2004; Gregor and Jones 2007) have contributed to the recognition of design science research as valid IS paradigm. Peffers et al. (2008) is an excellent example of cumulative tradition that belongs as well to this new school of thought. They motivate, present and demonstrate the use of a research methodology called 'design science research methodology', to which we adhere in this paper. This methodology distinguishes between five research activities: (1) problem identification and motivation, (2) definition of objectives for a solution to the problem (as defined by the research goal of this study), (3) design and development of the solution, (4) demonstration, which is concerned with the design, implementation and testing of a prototype that instantiates (i.e., "demonstrates") and validates the proposed solution, and (5) evaluation of the solution.

With this section covering research activities (1) and (2), the remainder of this paper is organised as follows. Section 2 provides an overview of carbon footprinting and carbon management software solutions. In Section 3, we discuss the most important logistics processes and the typical IS landscape of logistics service providers. Section 4 presents the main contribution of the paper, i.e., the design of a logistics carbon management reference architecture (research activity (3)) and its validation through a prototype (research activity (4)). The paper ends with conclusions and some pointers to future work (research activity (5)).

2 Carbon footprinting

2.1 Definition, scope and units

The term "carbon footprint" gained popularity as a measure of one's contribution to climate change. Basically, it is a derivation of the term "ecological footprint". While the ecological footprint is clearly defined (Wackernagel and Rees 1996), the origin of the term carbon footprint is unclear and its definition is ambiguous. In a column published in the New York Times, Wackernagel explains that the term was strongly boosted through a BP media campaign on the carbon footprint in 2005. The fact that the concept has emerged in and has been spread by media rather than by academia could be a reason for the lack of a clear definition. Wiedmann and Minx (2008) acknowledge this shortcoming and propose the following definition, which we also adopt:

"The carbon footprint is a measure of the exclusive total amount of carbon dioxide emissions that is directly and indirectly caused by an activity or is accumulated over the life stages of a product." The activity may be of individuals, companies, processes, etc. and the product can be a good or a service. The distinction between activities in general and product specific emissions is important, since two types of carbon footprints have emerged. The first reflects all emissions an organization is responsible for. It is the sum of all emissions directly or indirectly caused by the organization's activities. Indirectly caused emissions include, for example, office lighting and waste disposal. This high level footprint is called corporate carbon footprint (CCF). The second concerns the emission caused by a single product and is called product carbon footprint (PCF). Companies may calculate a footprint of a certain product to gain insight in their own processes, to promote the sustainable character of their products labelled with a carbon footprint, or to comply with European or governmental regulations. Finkbeiner (2009) mentions several other issues to consider when defining a carbon footprint, for example the scope of emissions, life cycle stages and other system boundaries, data sources and capital goods. A main issue in scientific debate concerns system boundaries, especially in the field of PCFs (Weidema 2008; McKinnon 2010; Matthews 2008). According to Wiedmann and Minx (2008) a PCF should take the whole life cycle into account, starting at design and manufacturing, via distribution to, and usage by the consumer, ending with the waste disposal.

For a footprint calculation one needs to know how much gas is emitted by performing a certain activity. However, for companies or individuals it is not feasible to reliably measure gas emissions themselves. Therefore specialised institutions perform field studies or experiments to determine standard values, which are called *emission factors* and are expressed as the weight of the pollutant per volume or weight of fuel substance (fuel-based - kg CO_2/litre), or per unit of distance (distance-based - kg CO_2/km) associated with the activity (Zadek and Schulz 2010). Fuel-based factors are the most reliable figures for carbon footprint calculation as they express the exact quantity of released CO_2. Distance-based factors can be used when fuel consumption data is not available. They use estimates or long term averages of fuel consumption, and, thus, are less accurate than fuel-based factors. For a global overview in the CCF of a company, this may suffice. But for detailed PCF calculations for an LSP, fuel-based factors are required.

2.2 Methods and standards

Two essentially different approaches on carbon footprinting can be distinguished (Wiedmann 2009). The first is a top-down approach based on input–output analysis. This is a technique modelling the relations between economic entities on macro level, e.g., related to industry sectors. When the total production level of a sector is known and statistics on total GHG emissions from that sector are available as well, one can allocate emissions to products. This technique is especially appropriate for CCF calculations. The second approach is bottom-up, based on process analysis. This method is used to quantify environmental impacts of individual products or processes and thus appropriate for PCFs. It is more detailed, but also easy to scope narrowly, taking only first-order processes and their impacts into account. A third, hybrid approach has also emerged. This allows for detailed calculation of core processes, while less significant processes, or activities elsewhere in the supply chain, can be covered by input–output analysis (Wiedmann 2009). The hybrid approach resembles activity based costing (ABC) and is referred to as activity based carboning.

Carbon footprinting is still a young and immature field, as also reflected by the co-existence of numerous standards, calculation methods, proposed by different organizations, and protocols (e.g., GHG protocol (Rich 2008), PAS 2050 (BSI 2008) and ISO 14064, 14069, 14040/14044 and 14067 standards, ARTEMIS (Boulter and McCrae 2007) and COPERT (Ntziachristos 2009) and (EEA 2009); HBEFA—The German Handbook Emission Factors for Road Transport; the Study on Transport Emissions of All Modalities (Boer et al. 2008)). Little synergy between them is visible, while consolidation would greatly improve the reliability, transparency and comparability of carbon footprints.

2.3 Green IS and existing CMSs

Within the IT sector two concepts emerged that are concerned with sustainability: Green IS and Green IT. Watson et al. (2008) define Green IS as the design and implementation of information systems that contribute to sustainable business processes. The difference between the two is that the first one sees IT as part of the problem, while Green IS contributes to the solution (Chen et al. 2009). This distinction is also explicit in the definition given by Naumann et al. (2011) to Green and Sustainable Software, a concept similar to green IS. Green IT typically improves IT energy efficiency, making the IT infrastructure itself more sustainable, and thus it reduces the problem. Green IS uses IT to make businesses "greener", and thus, creates solutions. Therefore Green IT relates to eco-efficiency and Green IS to eco-effectiveness. Several authors investigated why companies adopt Green IS and what their strategic considerations are (Chen et al. (2008), Mann et al. (2009), Molla (2008), Kuo and Dick (2010)). Kuo and Dick reviewed the extant literature and proposed a comprehensive adoption model, which incorporates organisational, motivational and technological factors found in various other theoretical models. A study that goes beyond the borders of one organisation, and investigates the factors impacting the sustainability of entire supply chains is presented in Ageron et al. (2011). Worth mentioning is also the Smart 2020 report (Webb 2008), in which the Climate Group quantifies the reduction opportunities Green IS offers, and identifies five categories of expected benefits from green IS adoption: standardisation (provisioning of standardised information or protocols; e.g., repositories with fuel emission factors), monitoring (incorporation monitoring systems in operations; e.g., emissions tracking), accounting (provisioning of a platform to account carbon emission; e.g., calculating auditable CCFs and CPFs), rethinking (offer innovative opportunities to change operations; e.g., route optimization, modal shift and driver behaviour), and transformation (application of integrated approaches to automate and change behaviour; e.g., chain wide emission tracking, cooperation to improve capacity utilization).

Another issue of relevance for this research concerns existing Carbon Management Software (CMS) packages, also called carbon accounting software. Although such an IS does not directly enable the reduction of carbon emissions, it does quantify them, providing a basis for reduction strategies. Its function is in the first place to monitor and account (Webb 2008).

In the remainder of this section we present the results of our survey of currently available CMS packages. The goal of the survey was to examine the functionality and sophistication level of such CMSs and to investigate whether any available package is suitable for the transportation industry. A few consultancy companies have recently investigated the CMS market. Most notably, the Verdantix' report (Verdantix 2009) positions 22 CMS vendors in a "magic quadrant", distinguishing between leaders, challengers, specialists and entrepreneurs. In another report, out of 60 investigated packages, Groom Energy Solutions (2010) lists eight leading vendors, which coincide more or less with Verdantix's leaders. Our own survey included all vendors that were listed as "leader" in one of the above-mentioned reports, a few "specialists" (Camco and CarbonView) and a few others. The selected CMS packages have been extensively compared on several functional and technological criteria (see Appendix A). We have drawn several conclusions from this tool comparison. It appears that most CMS vendors support footprint calculation according to different protocols and using various emission factor sources. Considerable differences in the level of sophistication were found concerning footprint calculation, and monitoring. Most vendors stick to corporate footprints, resulting in static (e.g., monthly) information. Others support product footprints as well, and are able to update the information more or less real-time. Looking at the decision support tools, they vary from simple target setting and progress tracking, to reduction plans with scenarios, financial impacts and benchmarks against other departments or companies. Finally, some packages also include accounting functionality. The most common deployment type is subscription to the CMS on a Software-as-a-Service (SaaS) basis. Other vendors sell their packages via the web or offer local installation as well.

Overall, the CMS packages of IHS/ESS, Hara and CarbonView can be regarded as the most sophisticated, with a wide range of functionality and some outstanding features offered on a SaaS basis. However, it is clear that all vendors target the generic company, emphasizing corporate footprint reporting and aiming for energy saving plans. Only CarbonView and Greenstone mention the integration with transport management systems. The latter also offers integrated distance calculation. This feature, however, does not make them qualify as specific for transport companies as they do not meet accuracy requirements for footprint calculations of transportation services.

3 A reference architecture for LCMS

Several definitions of a reference architecture exist in literature and practice. The World Wide Web Consortium (W3C 2004) gives the following definition: *"A reference architecture is the generalized architecture of several end systems that share one or more common domains. The reference architecture defines the infrastructure common to the end systems and the interfaces of components that will be included in the end systems. The reference architecture is then instantiated to create a software architecture of a specific system. The definition of the reference architecture facilitates deriving and extending new software architectures for classes of systems. A reference architecture, therefore, plays a dual role with regard to specific target software architectures. First, it generalizes and extracts common functions and configurations. Second, it provides a base for instantiating target systems that use that common base more reliably and cost effectively."*

Other interesting definitions have been proposed by Bass et al. (2003) and Greefhorst et al. (2009). Greefhorst et al. (2009) have compared various definitions and concrete examples of reference architectures and proposed the following definition: *A reference architecture is a generic architecture for a class of systems, based on best practices.* After comparing these definitions we infer that a reference architecture for a LCMS (1) is a generic architecture of the components of a system that provides functionality for the management of carbon emissions in the transportation sector; (2) provides a map of its possible relationships to and interfaces with both the business and the technological environment; and (3) is based on best practices.

The approach we took in order to design a reference architecture for LCMSs consists of the following steps. The *first step* is to get insight in the transport business domain, in order to understand what activities are carried out by LSPs and what emissions they cause. Therefore knowledge of LSPs' business processes is required. The *second step* is to gain insight in the IS landscape of an LSP. What information systems typically support the transport activities and what systems may be relevant as source of data for emission calculations? What interfaces are used to exchange this data? The *third step* is to unite the business and IS domains, to bridge the gaps and identify the relationships between them. The combination of these aspects forms the reference architecture. To the best of our knowledge this is the first attempt to design such an architecture for fuel-based transport emission calculation. Nevertheless, an LCMS should be also built upon best practices and, hence, should inherit the key features of generic CMS solutions available on the market.

For the development of the reference architecture specification, we comply to accepted enterprise architecture standards: we follow the development methodology prescribed by The Open Group Architecture Framework (TOGAF 2009), and we use the enterprise architecture modelling language ArchiMate (Iacob et al. 2011) for its formal specification. This choice has the advantage that ArchiMate is aligned with the (TOGAF 2009) methodology (as they are both international standards promoted by The Open Group) and they cover the same architectural domains.

3.1 Step 1. Transport business (process) domain

A transport is the movement of goods from one location to another. A shipment is a concrete quantity of freight. The party performing the transport is the carrier, also called logistics service provider (LSP). Its clients are shippers. The shipper orders the transport of a shipment and pays for it. For executing the transport, the carrier uses equipment and staff. A carrier may outsource (part of) the transport to a subcontracted carrier, called charter. A third party logistics provider is a business offering a variety of services, including warehousing and all kinds of freight handling.

Next we describe the transport process in more detail from the viewpoint of an order. Basically two types of orders can be distinguished: full-truckload and less-than-truckload. Often orders concern less freight than the full capacity of a trailer. To increase efficiency LSPs combine these less-than-truckload shipments into full-truckload ones. For the shipper the advantage is that the transport is cheaper. A drawback is that the transport takes more time, due to the fact that the route is broken down in intermediate legs between pick-up locations, trans-shipment facilities and drop locations. Three types of trans-shipment facilities can be distinguished: home depot (the depot from where a transport starts), hub (intermediate trans-shipment facility where shipments are regrouped and forwarded to common destination areas), and away depot (final trans-shipment facility a shipment passes).

In Fig. 1 the high level transport process is modelled (using ArchiMate). This process specification is based on literature (CapGemini 2007), board computer documentation, and interviews with Cape Groep[2] consultants and employees of transport companies. Therefore, next we explain to what extent emissions occur during process activities and we relate these to the PAS 2050 protocol (BSI 2008), which defines which emissions should be included in the footprint.

The transport process is triggered by the receipt of an *order*. An employee at the LSP's office registers the order, possibly after negotiating with the customer. As defined by the PAS 2050 protocol, the order receipt is the first step in the life cycle of a transport service, during which electric devices are used. Measuring the exact amount of electricity used for an order is very cumbersome. Therefore, these emissions are regarded as overhead and are allocated to a particular transport order using an activity based carboning method. This holds for all other similar office activities carried out during the process. Already at this stage, the shipper may desire a forecast of the emission caused by his order. The carrier could provide an estimation based on freight characteristics, distance to travel and expected equipment deployed for the trip. Such an estimation could be useful in case of a modal shift for a part of the route.

Planning a transport (also called trip) contains several sub-activities. The first is Load Planning. To maximize equipment utilization less-than-truckload shipments are consolidated to full-truckloads. Load planning assumes the elaboration of pick-up and delivery schedules, and may involve selecting the right transport modality (truck, train, air, or water).

The next step is to plan how the load will be moved. This can be done with own equipment or can be outsourced to another carrier (called charter). Thus tendering and carrier selection may be part of the process. When the LSP uses own equipment a planner has to allocate resources to the transport (i.e., specific equipment, driving and pulled unit, driver, etc.), and plan the actual trip. Based on the locations to be visited, now the exact route is planned, taking into account distances, legs, road types, and loading times. Next the *driver (or carrier) needs to be informed* regarding the trip plan. An employee collects all the trip details (i.e., vehicle to use, locations to visit, timeslots, shipment details,

[2] Cape Groep is a software company active in the logistics sector that participated in this research.

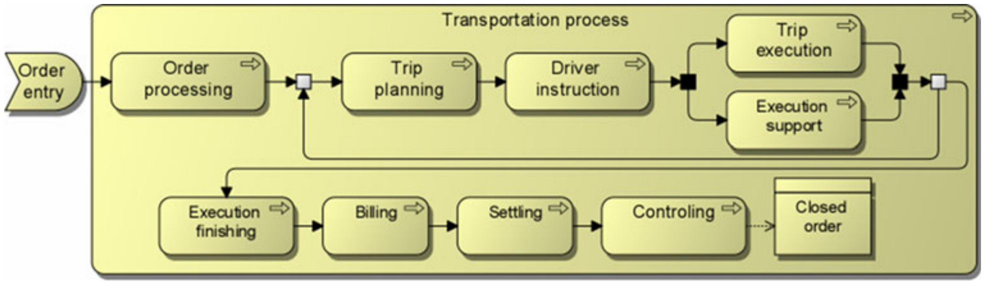

Fig. 1 High-level transport business process

additional activities to carry out), and any other relevant remarks and sends the instructions to the driver. Now the driver can *start his trip*. The first activity for a truck driver is to log on the board computer and identify himself. Subsequently the truck driver registers the mileage and relevant information on the vehicle status. Usually the truck is driven empty from the home base to the first pick-up location. This is considered part of a transport service and is therefore included in the footprint calculation. While driving, the consumption rate depends on the speed, road characteristics, traffic jams, load weight, etc. Fuel consumption can be determined as a combination of sensor data on fuel levels, coming from the motor management system, and board computer data on refuelling activities. When the driver arrives at the drop location he unloads the freight. After unloading the driver continues his route (if there still is freight in the truck to be delivered), or returns to the home base and logs off the board computer. The PAS 2050 protocol states that emissions caused during empty return trips (or intermediate legs) have to be included in the PCF. The trip *execution is monitored and supported* by employees at the LSP's office. If an exception occurs some action is taken (e.g., trip re-planning, driver's hours of service check-up, guarding security, calling/texting, and damage registration).

After the transportation has been executed, *financial settlement* with the shipper is required. If a load has been subcontracted, there is also a financial settlement with this carrier. An office employee consolidates relevant data (e.g., the time the transport took, the distance travelled, expenses made by the driver, proof of delivery, or other activities). He also determines the fuel consumption. For this the refuelling data entered in the board computer may be matched against fuel bills from the fuel supplier. From this data an invoice is created and sent to the shipper. If requested, the invoice may also contain the shipments' carbon emissions calculation. Therefore, at this stage the actual emissions of all activities associated with the shipment must be aggregated for the PCF calculation.

Controlling is not really part of the transportation, but it reviews the process. It calculates and accounts key performance indicators such as profitability, efficiency, and quality of the transport. Such measures typically include hours, speed, fuel usage, engine idle time, etc. Carbon emission may be added to the key performance indicators list. Based on aggregated data, reduction opportunities can be identified.

3.2 Step 2. Transport IS domain

This section gives an overview of the typical software applications landscape of logistic companies.

The overview is based on information gathered from Cape Groep consultants, and a large survey held by the professional association "Transport en Logistiek Nederland" among 450 of its members (TLN 2008). The results reported in (TLN 2008) indicate that the presence and complexity of the IS landscape grows with the fleet size. Therefore, it is realistic to expect that a LCMS is most relevant for larger LSPs, which have complex emission monitoring and would benefit the most from reduction strategies. In Table 1 we briefly present some of the systems that make up the logistics IS landscape, and that may serve as source systems for data used by a LCMS. The context diagram shown in Fig. 2 depicts these systems and their relationships with a LCMS. More exactly, it shows all information systems that may be present in the environment of the LCMS and summarizes what data they may exchange with the LCMS. The model abstracts from how this data exchange takes place.

When (a part of) a transport is subcontracted, data on the distance travelled or even on fuel consumption has to be delivered by the charter (by integrating the charter's BC or TMS, with the LCMS or in some other way; e.g., an invoice, personal contact, etc.). The LCMS has to distinguish between trips executed by own equipment and by charters in order to differentiate on the various scopes of a carbon footprint. Furthermore a mapping system may be consulted to determine the distance between two locations, which might be necessary for the calculation of emissions resulting from an order.

3.3 Step 3. Reference architecture for transport carbon management IS

Since we have investigated the business process and information systems of an LSP separately, the next step is to integrate these domains (i.e., business and IT), and examine

Table 1 Description of logistic information systems and their relation with LCMS

System's name	Main functionality	CMS relevant data
Order entry system (OES)	Shippers can order transports from carriers via Internet. 40% of LSPs with more than 50 pulling units are using automated order entry and another 40% intends to implement it.	Freight characteristics, like volume and desired condition, and time restrictions for the pick-up and delivery.
Transport management system (TMS)	Supports several critical processes of an LSP and is therefore widely used. Over 90% of large carriers have implemented a TMS. It forms the administrative heart of an LSP. Main functionality: - Stores relations with shippers and subcontractors, quotes, order and tariff arrangements; - Files & registers orders (whether integrated with an OES or entered manually by an employee) - Supports initial load planning, and sometimes also transport planning. In most solutions however, a planner has to finalize the planning, or it is supported by an APS (see below). - Covers automated carrier selection. - Creates the transport job and dispatches the job to the scheduled driver. - For supporting the trip execution, communicates with board computers. It sends jobs to drivers and receives progress information, tracing events such as arrival, loading, unloading. - Supports the billing and settlements of transports. - Provides management information measuring and reporting key performance indicators.	Route details (e.g., like locations to visit, and which shipments are combined in the truck). The emissions of a certain shipment should be delivered to the TMS, so it can be stored and presented on the invoice.
Advanced planning system (APS)	APSs are used in large transport with high amounts of a large fleet and many constraints to cope with, where the transport planning is often too complex for a TMS or a human planner. About 60% of the large LSPs (over fifty driving units) has implemented an APS. - Supports all the facets of transport planning, from load and fulfilment planning to load design and finally trip planning. - Monitors the transport real-time and is able to update the planning if necessary. - The system is usually integrated with the TMS, but some solutions directly send schedules to BCs.	Route and shipment details
Board computer (BC)	BCs enable a real-time exchange of information between home-base and trucks and are mostly integrated with the TMS. 95% of the large transport companies and over 50% of medium ones are using board computers. BCs typically have three main functions: - Two-way communication between truck (driver) and office (IS/employee): based on existing communication protocols (GSM module allows for communication via SMS text messages or speech, GPRS or UMTS). Data packets can be sent and received, typically in XML format. When its trucks often visit remote areas, an LSP may opt for support of communication via satellite.	For calculating carbon emissions of transports, the combination of board computer and CANbus is most relevant. At each activity entered in the BC, like unloading, the current fuel level is recorded by the CANbus and communicated to the CMS. This way the fuel consumption for each part of a trip can be calculated and allocated to the shipments involved.

Table 1 (continued)

System's name	Main functionality	CMS relevant data
	- Data registration of manually entered data on the progress of the trip execution such as activities (e.g., like arrival, waiting, unloading, refuelling, departure, etc.) and details about the activity (e.g., number of pallets unloaded, amount of diesel fuelled, proof of delivery by digitally signing on the screen etc.), which is sent real-time to the home base of the LSP, where the transport can be monitored. Starting and finishing activities by the truck driver are used for the registration of the hours of service by the BC. The BC also records all kind of data automatically, (e.g., mileage). The BC can have an interface with the motor management system or CANbus (Controlled Area Network bus, a standard for communication between microcontrollers and devices within a vehicle with host computer). Thus the BC can record truck and driving behaviour data (e.g., speed, acceleration, braking, fuel level). Fuel consumption is recorded and checked to improve drivers' driving behaviour or to detect anomalies at refuelling (e.g., fuel theft). - Localization and navigation: Via GPS (Global Positioning System) the exact location of the truck is always known. The position is continuously sent to the home base, supporting tracking and tracing as well as planning. BCs contain often navigation software.	
Accounting information system (AIS)	Is a general accounting system, storing all corporate expenses and revenues. This generally includes utility data, which means that this IS can provide data on the consumption of electricity, gas and water at premises of the LSP. Ideally smart energy meters track consumption continuously, or for example monthly.	This data cannot be directly related to transport services, but must be used for the calculation of corporate carbon footprints.
Warehouse management system (WMS)	Controls the movement and storage of goods in a warehouse. Furthermore it tracks the receiving and shipping of items.	A WMS acts as a source of information for the CMS in two ways. It helps to keep track of shipments that are transported over several hubs and it provides stocking information.
Fleet management system (FMS)	Stores data on the status of the carrier's vehicles, both driving and pulled units. First, it contains all the basic and factory data of trucks, such as the license plate, construction year, engine type, fuel type and average fuel consumption as reported by the manufacturer. Second, it maintains historical data on the mileage of trucks, actual fuel consumption, and maintenance performed on the vehicle. Third, it provides functionality for vehicle tracking & tracing using GPS.	Vehicle data
Cooling unit sensor	In case of refrigerated transport, a sensor system could be implemented in the cooling unit. It sometimes communicates with the board computer, but may also send its data directly to (an IS at) the home base via a wireless connection like GPRS.	Advanced sensor systems monitor the fuel consumption due to the refrigerator unit, which is used for fuel-based PCFs.

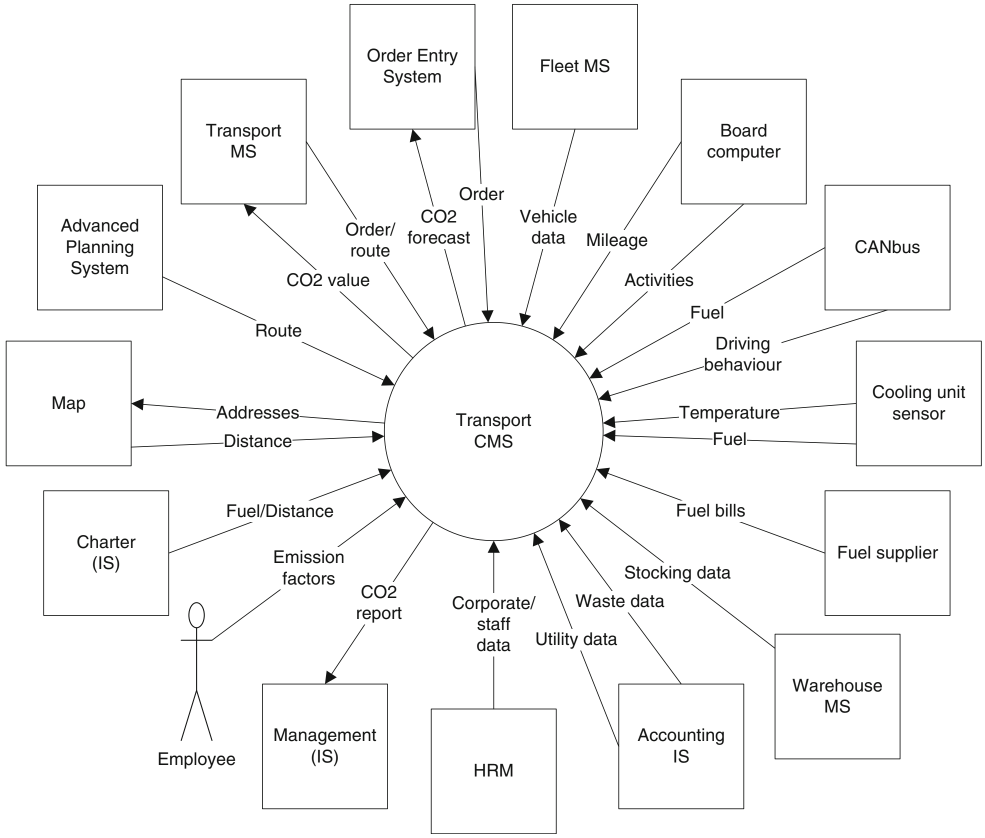

Fig. 2 Context diagram of a Transport CMS

how the transport PCF can be calculated. For a transport service, a carbon footprint is considered useful in two occasions during the transport process. First, a planned value as an emission forecast is calculated when an order is processed. Second, an actual value (i.e., the transport PCF) is produced, after the transport has been executed. The following paragraphs summarize the key application functions necessary to produce forecasted and actual footprints.

Furthermore, based on best practices in the CMS market we identify key features useful for the LCMS and the logistic sector as well. Finally, we present our LCMS reference architecture (modelled in ArchiMate), divided into the business, application and technology layers, as prescribed by TOGAF and ArchiMate.

Transport PCF key functionality The first key functionality is to provide a *forecast* of the emissions caused by a transport order. A carrier can use such a forecast to differentiate from competitors or, internally, to review routing alternatives including a modal shift. A generic approach to determine an emission forecast is to build it from the following data:

- Order data, including freight characteristics, locations and times (provided by an OES).
- Vehicle data, including probable equipment to be deployed and its historic fuel consumption average (provided by an FMS).
- The route to be travelled, including various scenarios for hubs and modal shifts (provided by a TMS, a map system, or a database with various routes between known locations).

The planned value can be calculated as a weighted average for several vehicle and routing alternatives. Norms could be determined from historic data for vehicles, routes and actual emission values calculated for past transports. The combination of cargo, vehicle and distance leads to an expected amount of fuel consumption. Finally the emission factors corresponding to the sources to be used need to be collected. Since many requests for emission factors are to be expected, it appears to be the best option to include such a database in the CMS. The forecast is then calculated by multiplying the expected fuel consumption with the corresponding emission factor.

The second key functionality is to provide support for the calculation of the transport *PCFs*. As opposed to a forecasted value, a transport PCF is based on actual values and thus calculates the actual emissions after a transport. The challenge here is to harvest all necessary data and both break down and sum up these data to a single PCF. A wide range of activities has to be taken into account and data may come from several ISs, some even in real-time. A generic approach to calculate a transport PCF is to build it from the following data:

- Order data, including freight characteristics (provided by an OES).
- Vehicle data, including the actual driving and freight units deployed for the transport (provided by a TMS).
- Activity data, while executing the trip (provided by the BCs and the WMS).

For a dedicated single truck road transport, a PCF can be calculated from this data by multiplying the amount of energy used with the corresponding emission factor. However, often shipments are combined and transported over multiple legs by various vehicles, sometimes by charters and/or using multiple modalities. Therefore a CMS should also have the functionality to store intermediate activities and emissions of single legs, allocate activities/emissions to shipments combined in the leg, receive data on subcontracted legs from charters (generally via the LSP's own TMS), aggregate intermediate values from several legs and calculate the shipment's total footprint.

Best practice key functionality The third key functionality is to *maintain emission factors*. The forecast and PCF calculations rely on accurate and up-to-date emission factors, which are often requested by the CMS. These factors are regularly renewed due to improved measurements and changing policies. The survey of the CMS market showed that most solutions support a variety of footprint protocols and emission factors. As LSPs often operate internationally and sometimes combinations of fuel-based and distance-based calculations are required (in case of charters) maintaining an emission factor database is desirable.

The fourth key functionality is to provide *CCFs*, in standard reporting formats. Besides being able to provide PCFs to shippers, a carrier will also wish to gain insight in its own total footprint. CCF is the most basic CMS functionality supported by all CMS solutions. To arrive at a CCF, the CMS needs to collect relevant data from the AIS, like energy used in a certain period. For more specialized CCFs the CMS may connect to a human resource management system for department-level footprints or aggregate PCFs from transports in a certain period.

The fifth key functionality is to *support reduction strategies*. After gaining insight in its own emissions, an LSP would probably want to reduce them. Offering assistance in this process is a must, as the CMS market survey revealed. The existing solutions all have their own approaches on taking action: manual or automated data analysis, simple targets or including benchmarks and scenarios, and different ways of tracking progress. But altogether reduction support is a clear best practice. Existing solutions usually take CCFs as a starting point and define targets for future periods, for a certain energy source or department. In logistics PCFs can serve as starting point as well. Suggesting modal shifts for similar transports in the future or driving behaviour analysis from CANbus data are examples of reduction strategies.

The five key functions for a transport CMS form the basis for our proposed reference architecture, shown in Fig. 3. In the sequel, the elements of the architecture are briefly discussed, layer by layer.

Business layer The upper layer of the architecture shows the most relevant actors for the transport CMS. These are: the shipper, the carrier, and the charter, which is a *specialization*[3] of a carrier. Obviously, more actors perform a role in the transport process but for the sake of simplicity we abstract from them. The external business services the actors *use* from the CMS are emission forecast (i.e., "Planned value service"), transport PCF (i.e., "Actual value service"), Corporate CF and reduction plan. Another part of the business layer is the LSP's business process which was explained in detail earlier.

Application layer This layer contains the CMS application components, the CMS' IS environment and the services it delivers. The upper part of the application layer consists of external application services. These are *used* by different activities of the business process and are represented as system outputs. These services and the application components realising them generally constitute the CMS. The five modules depicted in Fig. 3 each implement one of the key functions described earlier. Since the corporate footprint module cannot be seen as innovative, we do not specify this module in detail. The reduction module is also not further detailed since reduction initiatives may be very diverse, and organisation specific. This would make the reference architecture less generic. Four of the modules *realize* each one external application service. The protocol module, however, *realizes* an internal application service, namely the emission factor service. This service is *used* by all other footprint modules to collect emission factors. From the emission calculator module, data may *flow* to other modules, e.g., to set norms for forecasts, to aggregate PCFs in a CCF, or to serve as input for reduction plans. The application layer also contains services realized by relevant ISs in the CMS' environment. These services are all *used* by some of the CMS' components.

Infrastructure layer The third layer represents the infrastructure of the CMS and its environment. It shows the hardware, software and network deployment of the services that are supported by the infrastructure. The first node in the infrastructure layer is the CMS server running the applications performing calculations, generating reports, etc., and including a database management system *assigned* to (i.e., deployed on) the application server. The DBMS (data base

[3] In the remainder of this section the words written in italics designate relationships between architecture elements from Figure 3.

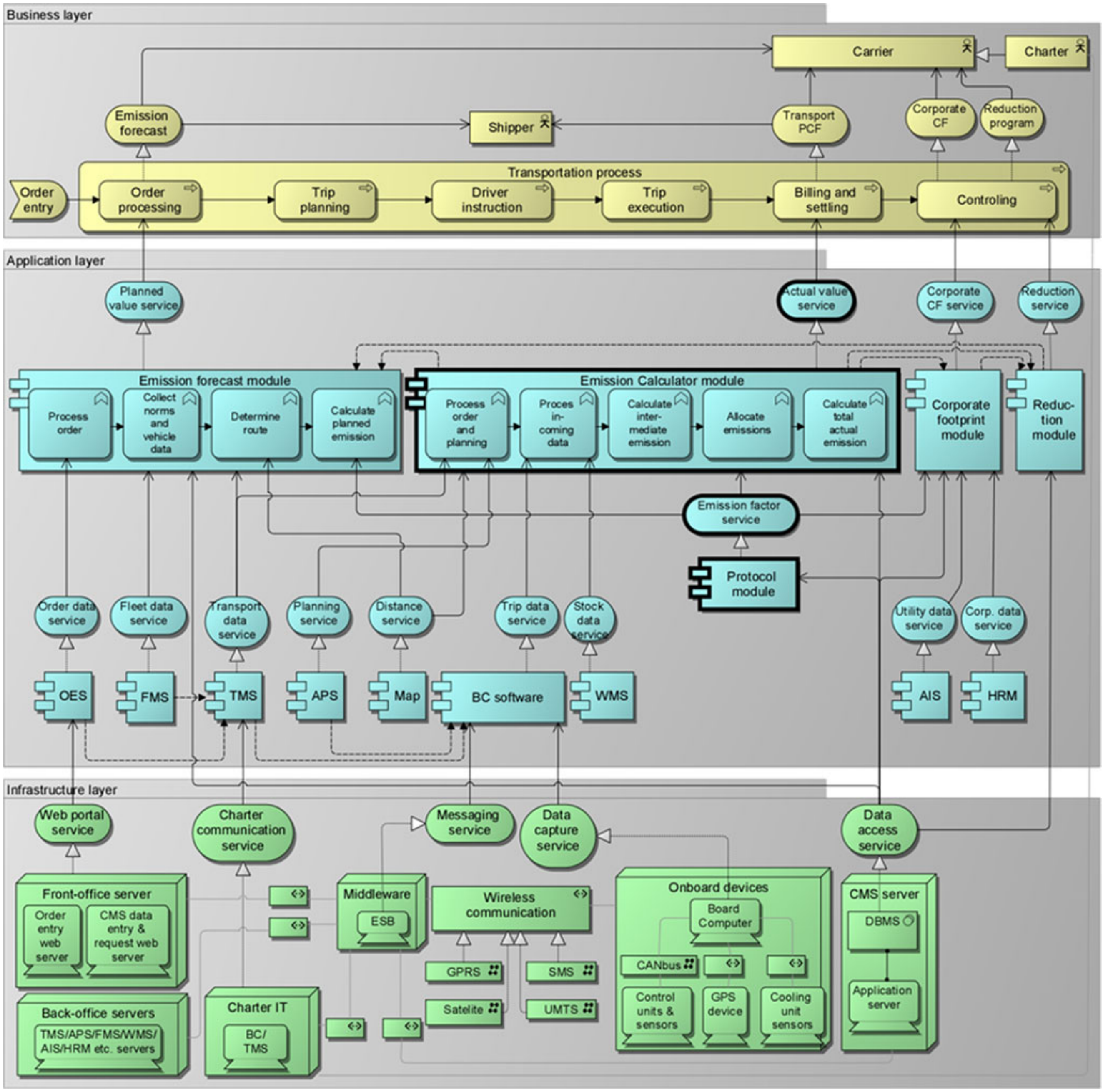

Fig. 3 Reference architecture for a transport carbon management system

management system) *realizes* a data access service (of course including create, read, write and update operations) that is used by the CMS application components. In the environment of the CMS server both front-office and back-office servers are present, as well as on-board devices. These are all connected to each other via middleware, using communication paths that *associate* the nodes to each other. We opt for this design as it is now current practice, as also shown by the CMS market survey, to provide web-based service oriented architectures. Most communication paths have not been specified, they are typically wired networks.

A front-office server *realizes* a web portal, which is *used* by the order entry system. The order entry web access, depicted as artefact, is the start of any transport process. Furthermore, CMS users access the CMS information system via the front-office server when they want to enter data or request information. The reference architecture abstracts from where the nodes are located. As long as both carriers and shippers can access the CMS, it can be deployed web-based at the carrier or the CMS supplier sites, or as SaaS solution at the CMS supplier site or at an external hosting provider site. The latter is an important requirement, as SaaS is the most common deployment model for carbon management software, and the logistics industry is starting to adopt SaaS solutions as well (e.g., Holtkamp et al. 2010). The other ISs shown in the application layer are aggregated in the Back office servers node. Details on the deployment of these ISs are out of scope of the reference architecture. On-board devices are important, as they enable fuel-based carbon footprint calculations, as explained earlier. Thus, board computers *realize* a data capture service, *used* by the BC software. BCs also connect to other on-board devices (e.g., via the CANbus network to the motor management system). Most important for the CMS is that data exchange over GPRS[4] or UMTS,[5] is formatted in XML. The middleware node represents a hub-and-spoke solution to allow communication between all other infrastructure nodes. An Enterprise Service Bus could implement such a node.

Considering its central role, the CMS server is depicted separately, although it may be just as well one of the front or back-office servers. In fact, although the architecture may show various servers as different nodes, in practice these may be virtual nodes deployed on a single server node. In general, the reference architecture does not prescribe how to deploy the servers, excludes telematics issues like network characteristics (e.g. routers) and does not take qualitative requirements into account, like security issues.

4 CMS design and prototype

This section presents a prototype for the Product Carbon Footprint Calculator module of the LCMS. The goal of implementing the prototype is to verify if an IS can be built according to the reference architecture and to validate whether such an IS is able to provide correct fuel-based carbon footprints of transport services. We first explain the scope of the prototype, followed by a detailed discussion of the prototype design, covering the domain model, the interfaces with the TMS and the BC, and the carbon footprint calculation functionality. A test case for the prototype is performed, after which we reflect on the verification of the calculated values, and on the limitations of this prototype.

4.1 Scope, development platform and development methodology

Of the five main application components identified in the application layer of the LCMS reference architecture, the prototype only implements the Emission Calculator module and a simplified version of the Protocol module. The decision to limit the scope of the prototype to these two modules was motivated by the fact that the other components are either implemented in existing products (e.g., the CCF module), or they are not relevant for illustrating the proposed PCF fuel-based approach (e.g., the Reduction module).

The architectural entities depicted with a thick border in Fig. 3 define the architecture of the prototype, and set its scope by exactly showing which parts of the reference architecture have been selected for implementation. The very core of the prototype is the Emission Calculator module (in other words, the PCF calculator). It contains (complex) functionality most relevant to logistics and interacts with several systems present in the logistics domain.

The prototype was realised using the Mendix Business Modeler (Mendix 2011). Mendix is an agile, model-driven application development platform that facilitates easy integration of existing applications and newly developed ones. The choice for Mendix had practical reasons, since our prototype was supposed to extend and complement a PCF forecasting module that the Cape Groep developed using Mendix, prior to this research. The choice for Mendix has also the advantage of being integrated with software support for the agile software development methodology SCRUM,[6] which is the development methodology we followed when developing the prototype. Mendix requires the specification of three types of models from which the future application software is generated in a model-driven fashion: a *data model* (which resembles closely UML[7] class diagrams), the *application behaviour model* (i.e., the application logic), specified in terms of so-called "microflows" (for which a subset of the BPMN[8] notation is used), and the *specification of the (user) interfaces*. In the following sections, a few such models are shown and explained. As far as the infrastructure layer of the reference architecture is concerned, we design the prototype as a loosely coupled application that exchanges data over the web in standard XML format.

Finally, since our prototype fits in the definition of green and sustainable systems, we can position it (in terms of lifecycle) with respect to the GREENSOFT reference model (Naumann et al. 2011) as covering GREENSOFT's development phase.

4.2 Prototype design—data model

The data model for the PCF Calculator module is shown in Fig. 4 and has been harvested from Mendix (as all other models in this section). It should be noted that a similar

[4] General Packet Radio Service (GPRS).
[5] Universal Mobile Telecommunications System (UMTS) is a third generation mobile cellular system for networks based on the GSM standard.

[6] www.scrum.org, http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum_Guide.pdf
[7] Unified Modeling Language (UML), www.uml.org.
[8] Business Process Modeling Notation (BPMN), www.bpmn.org.

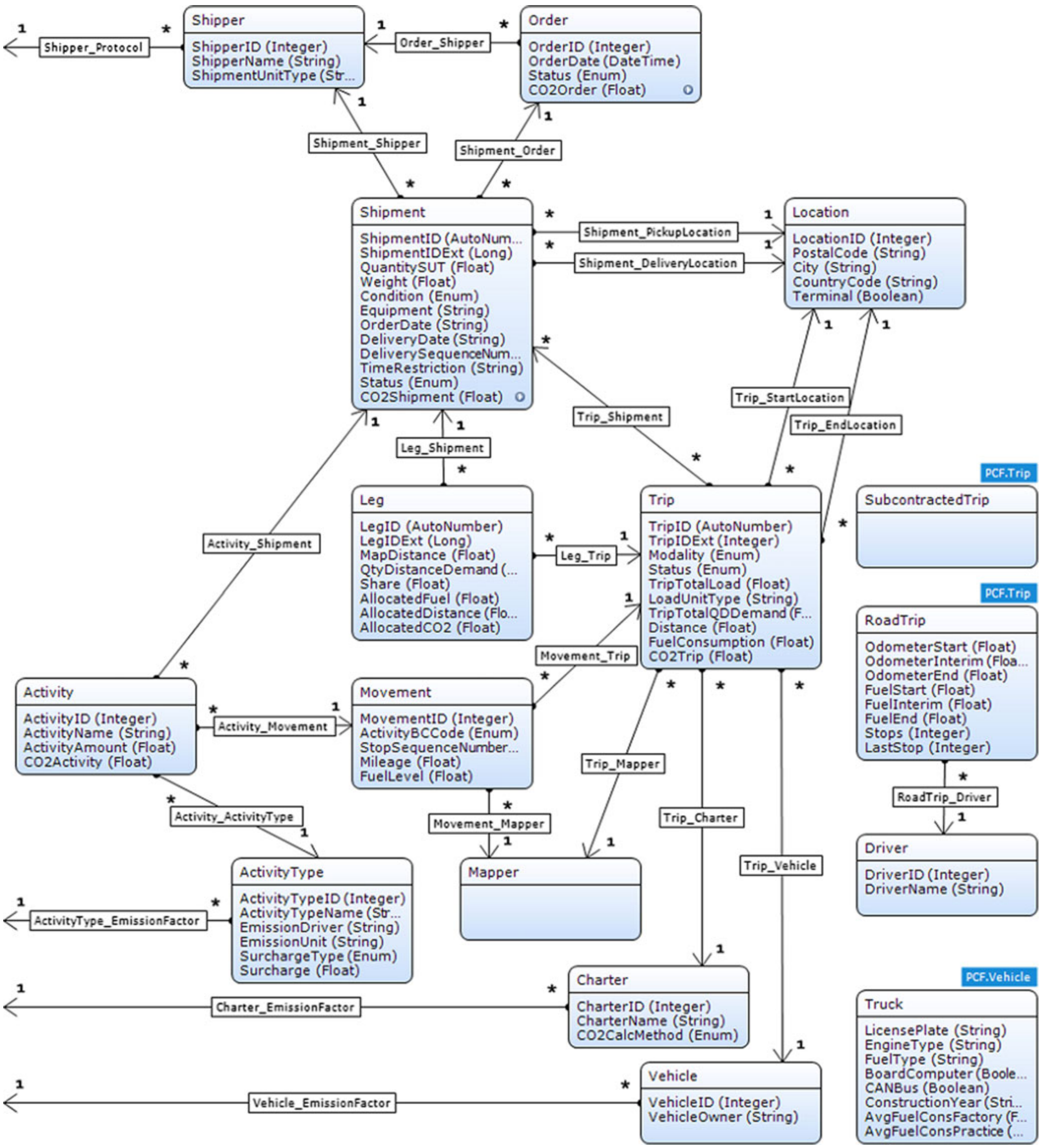

Fig. 4 Data model of the Product Carbon Footprint Calculator module of the prototype

model has been defined for the Protocol module. The model defines the various data entities (their attributes and relationships) manipulated by this module. Due to space limitations we explain here just one of these entities, as an example. The central entity, *Shipment*, contains shipment details and the carbon emission caused by this shipment, which is calculated after all *Trips* in which the shipment has been involved have been finished. For incoming shipments the prototype assumes the quantity is provided in the *shipper*'s default shipment unit type and by the shipment's weight. Furthermore its required temperature condition, equipment during transport (truck-mountable forklift, loader crane, etc.), and transport dates are stored. Finally the transport status is also tracked in this object. One or more shipments are in practice usually part of a transport *order*. Also, for practical reasons the prototype also links them directly to the *shipper*. Shipments are assigned an automatically generated number as internal ID. This is mainly done to ensure uniqueness and easy sort on the creation moment.

4.3 Prototype design—interfaces

The prototype was designed under the assumption that the PCF calculator must interact with a TMS application, from which it receives information regarding trips planning, and with a BC, from which it retrieves real-time updates on trips in progress. These two interfaces are included in the prototype, and will be briefly discussed in the remainder of this section. The calculator module uses these interfaces to acquire data for its first two core activities. "Process order and planning" handles planning data of incoming trips and "process incoming data" handles incoming messages from board computers (see Fig. 3).

First we have specified an XSD[9] that consists of the elements a trip planning XML file should contain and defines the way the elements are structured in the XML file. After importing the XSD file, in Mendix, it is possible to map its elements onto data objects and attributes in the PCF data model. Thus, complete trip plannings can be mapped to domain objects automatically.

The import of BC data that provide trip progress information and actual fuel consumption data, has been implemented in a similar fashion.

4.4 Prototype design—PCF module application logic

This section focuses on the design of the application logic for the footprint calculation. First we show how we calculate the emission of a complete trip, and second, we explain how we allocate emissions to combined shipments and aggregate various emissions of a certain shipment to a final value.

Calculate trip emission The third core function of the PCF calculator module as defined in the reference architecture is to calculate intermediate emissions. It is up to the designer to decide which intermediate emissions are determined. One could go for calculating emissions for each movement. For the prototype we chose a complete trip as the intermediate calculation, since this is sufficient for validating that the carbon emissions of a transport can be calculated fuel-based using board computer and CANbus data. Figure 5 shows the microflow that calculates the carbon emissions of a trip. With a trip as input, it first determines the trip type, either road or subcontracted. Then the microflow splits depending on the trip type. This microflow also looks up the emission factor that fits with the truck's properties. But by assigning BC and CANbus to each truck, the retrieved emission factor is always fuel-based in litres of the truck engine's fuel type. By multiplying the emission factor with the fuel consumption previously stored as road trip attribute, the microflow calculates the carbon emission of the trip. In case of subcontracted trips, the model assumes that charters provide distance travelled only. The microflow looks up the emission factor at the associated charter object, and calculates the emission of the trip as the product of the emission factor, distance and load.

Allocation and aggregation of emissions In the reference architecture the fourth core function of the emission calculator module is to allocate emissions to the shipments combined in the trip. The allocation takes place in the leg object that is associated to each shipment-trip combination. After the carbon emission caused by a trip has been calculated, it is rather straightforward to execute the allocation. The microflow that is responsible for the allocation receives a trip as input, and then performs a database query to receive all legs associated with the trip. Next, while iterating over the list, it calculates the allocated amounts of distance travelled, fuel consumption (both with the purpose to provide extra information), and, most important, the carbon emission.

Finally, the fifth core function declared for the emission calculator module is to calculate the total actual emission for a transport service. This means accumulating the emissions allocated to all legs over which the shipment has been transported from pickup location to delivery location (Fig. 6).

Furthermore, emissions caused by other activities assigned to the shipment, e.g., surcharges for fuel consumption by forklifts in warehouses, or by refrigeration units, have to be added to the transport product carbon footprint.

4.5 Prototype test case and evaluation

After explaining the functionality of the prototype, we now demonstrate its functioning with a simple test case. In Fig. 7 a multi-stop less-than-truckload road trip (including the allocated fuel consumption, c_i, for each shipment i) is shown.

Figure 8 shows a screenshot of the result of the trip planning data import and of the corresponding microflow execution. The Trips table displays the imported trip (number 23051) that contains a list of the four shipments. Details on these shipments are presented in the Shipments table. It provides freight and order information (without showing all location details). Finally the Legs table lists the legs created

[9] XML Schema Definition (XSD), http://www.w3.org/XML/Schema.html.

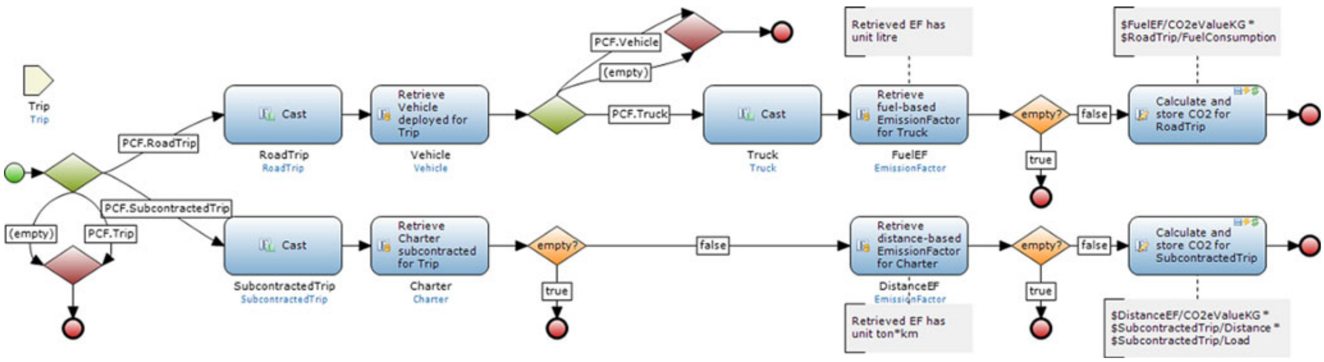

Fig. 5 Microflow to calculate the carbon emission caused by a trip

for each shipment-trip combination (the definition of all these tables can be found in the data model from Fig. 4). After the trip planning has been received, it is a matter of waiting for incoming messages from the board computer of the deployed truck (or updated trip planning data from the TMS).

Figure 9 shows the tables after all movements have been completed and the trip has been finished. The Movements table gives an overview of all movements and how the mileage and fuel level developed during the trip. The Road Trips table shows the same trip as the previous picture, but now its specialization containing all the odometer and fuel details. Based on the fuel consumption the carbon emission for the trip is calculated and stored. In the Shipments table, where the prototype does not change the delivery date or time, the carbon emission for each shipment is now provided after the trip has been finished. The source for these values is the Legs table with the associated leg objects. Based on the shares, the allocated distance, fuel consumption and carbon emissions have been determined.

Although this test case is limited to just one round trip, it does prove that four of its intended core functions have been performed correctly. Furthermore, it shows how the reference architecture can be instantiated in an application that can correctly calculate fuel-based transport carbon emissions using BC data.

The test case shows that the prototype is able to calculate the emissions for a single transport service. However, large transport companies operate a fleet of hundreds of vehicles executing many trips each day. This results in a huge amount of incoming board computer data. This raises the question whether calculating emissions real-time for all those transport jobs is feasible in practice. The real-time handling on a large scale of BC data is not a problem since such systems are currently implemented in many LSPs and feed data real-time in their back-office TMSs. This data can be made available to a CMS as well. The next question is whether the CMS is able to handle large amounts of incoming messages and to perform subsequent calculations. Experiences during past projects with BC communication revealed that implementing such applications is feasible. Similar requirements were posed (and could be met), in the case of a track-and-trace application, that calculates estimated times of arrival based on activities reported from BCs on trucks. The hardware deployed for this application is modest, running on a single server with quite ordinary specifications. Furthermore, a property of the Mendix platform is that it supports load distribution over multiple machines. Thus, in case the carrier's fleet becomes very large, or calculations become more complex, the application remains scalable. Based on these experiences we conclude that scalability is not a major issue when the prototype is extended to an actual commercial product (which is currently work in progress).

5 Conclusions and future work

As one of the few sectors where carbon emissions have risen over the last decade, the logistics industry is under pressure

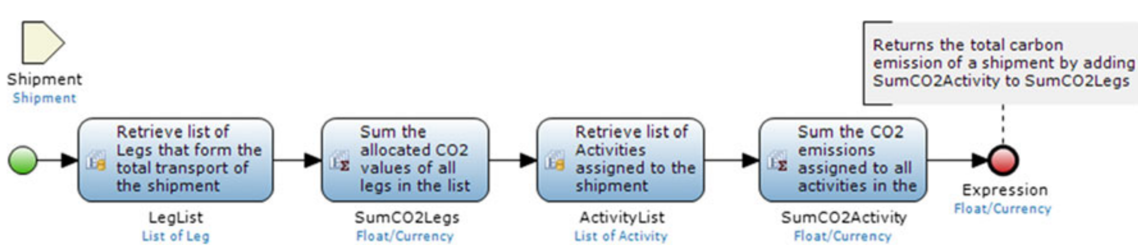

Fig. 6 Microflow to calculate the total carbon emission of a shipment caused by the transport service

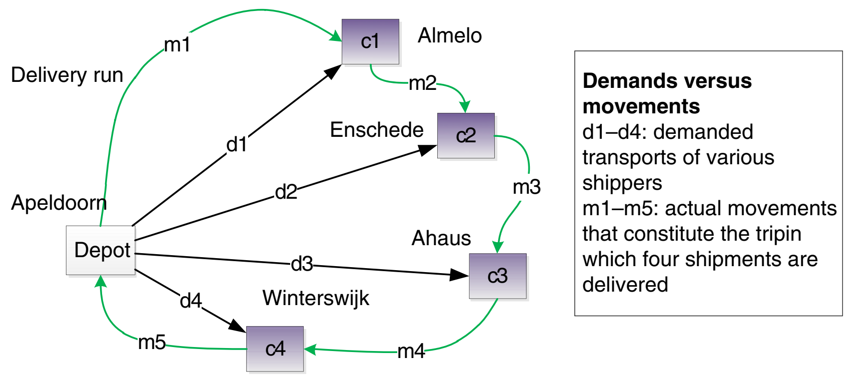

Fig. 7 Demanded transports versus actual movements of a trip used to test the prototype

to become more energy-aware and -efficient. A carbon management information system is an instrument that can help to achieve this. However, our market survey among CMS vendors reveals that no existing CMS software qualifies for accurate product footprinting of transport services. In order to provide a solid basis for the design of suitable carbon management information systems for LSPs, we propose a reference architecture that covers the following main key functions: forecasting of carbon emissions caused by a transport order, fuel-based calculation of transport PCFs by obtaining fuel usage of vehicles from board computer data, maintaining emission factors, calculation of CCFs in standard reporting formats, and support for reduction strategies. This reference architecture was instantiated and validated by means of a prototype. A test consisting of a simulation of a roundtrip with multiple shipments has been carried out.

We conclude this study with a discussion of several *limitations* of this work. As indicated below, these limitations give rise to possible extensions, and improvements of the reference architecture, or of its prototype implementation, as future work.

Empirical validation The prototype showed that core ideas behind the reference architecture are correct and feasible. Nevertheless, this prototype should be tested in more realistic settings, using more complex test scenarios (e.g., more trip types, larger fleet, more companies involved, etc.), in order to quantitatively assess its performance.

PCF data analysis and mining Over time many PCFs are calculated. The ultimate goal of gathering this data is to find reduction opportunities. Further research is necessary, using for example data mining techniques, to explore this data and to determine which data is useful to base reduction programs on. However, this type of research is possible only when a large amount of historical data has been collected, which is not yet the case. Such research should result in a concrete design of the reduction module, which we have left unspecified for now.

Comparison of fuel-based and distance-based outcomes In line with the previous limitation, another interesting type of analysis left unexplored concerns a large scale comparison

Trips

Search New Edit Delete 1 to 1 of 1

Trip ID	Shipments	Status	Modality	Truck	Charter	Load	Unit	LoadDistDmnd	Distance	Fuel (l)	CO2 trip
23051	44662;44663;44699;44714	Planned	Road	BZ-ZZ-22		13.00	LM	1015.00	0.00	0.00	0.00

Shipments

Search New Edit Delete 1 to 4 of 4

Shipment ID	Shipper	Quantity	Unit	Weight kg	Temp	Equipment	Order date	Delivery date	Time	Stop #	Area	Status	CO2 kg
44662	Kamp Electro	4.00	LM	0.00	Ambient		09-03-2011		11:30	1	7605QW	Planned	0.00
44663	Kamp Electro	1.00	LM	0.00	Ambient		09-03-2011		12:00	2	7533ER	Planned	0.00
44699	Grum GMBH	5.00	LM	0.00	Ambient		09-03-2011			3	48683	Planned	0.00
44714	Vries BV	3.00	LM	0.00	Ambient		09-03-2011		17:00	4	7101TY	Planned	0.00

Legs

Search New Edit Delete 1 to 4 of 4

Leg ID	Shipment	Trip	Qty	Unit	Kg	Map distance	Qty Dist Dmnd	Share	Truck	Charter	Allocated fuel	Distance	CO2
230511	44662	23051	4.00	LM	0.00	57.00	228.00	0.22	BZ-ZZ-22		0.00	0.00	0.00
230512	44663	23051	1.00	LM	0.00	69.00	69.00	0.07	BZ-ZZ-22		0.00	0.00	0.00
230513	44699	23051	5.00	LM	0.00	98.00	490.00	0.48	BZ-ZZ-22		0.00	0.00	0.00
230514	44714	23051	3.00	LM	0.00	76.00	228.00	0.22	BZ-ZZ-22		0.00	0.00	0.00

Fig. 8 Combined screenshots of tables with the Trip, Shipments and Legs after import of the trip planning

Movements

Search New Edit Delete 1 to 6 of 6

Movement ID	Trip ID	Activity ID	Stop #	Mileage	Fuel level
723612	23051	Start trip	0	675839.00	664.00
723698	23051	Unload	1	675901.00	647.00
723748	23051	Unload	2	675926.00	641.00
723795	23051	Unload	3	675952.00	634.00
723888	23051	Unload	4	675981.00	626.00
723967	23051	End trip	5	676061.00	604.00

Road Trips

Search New Edit Delete 1 to 1 of 1

Trip ID	Truck	Odo start	Interim	Odo end	Distance	Fuel start	Interim	Fuel end	Consumption	Status	Stops	Last	Load	Total LMD	CO2
23051	BZ-ZZ-22	675839.00	142.00	676061.00	222.00	664.00	38.00	604.00	60.00	Finished	4	4	13.00	1015.00	188.40

Shipments

Search New Edit Delete 1 to 4 of 4

Shipment ID	Shipper	Quantity	Unit	Weight kg	Temp	Equipment	Order date	Delivery date	Time	Stop #	Area	Status	CO2 kg
44662	Kamp Electro	4.00	LM	0.00	Ambient		09-03-2011		11:30	1	7605QW	Finished	42.32
44663	Kamp Electro	1.00	LM	0.00	Ambient		09-03-2011		12:00	2	7533ER	Finished	12.81
44699	Grum GMBH	5.00	LM	0.00	Ambient		09-03-2011			3	48683	Finished	90.95
44714	Vries BV	3.00	LM	0.00	Ambient		09-03-2011		17:00	4	7101TY	Finished	42.32

Legs

Search New Edit Delete 1 to 4 of 4

Leg ID	Shipment	Trip	Qty	Unit	Kg	Map distance	Qty Dist Dmnd	Share	Truck	Charter	Allocated fuel	Distance	CO2
230511	44662	23051	4.00	LM	0.00	57.00	228.00	0.22	BZ-ZZ-22		13.48	49.87	42.32
230512	44663	23051	1.00	LM	0.00	69.00	69.00	0.07	BZ-ZZ-22		4.08	15.09	12.81
230513	44699	23051	5.00	LM	0.00	98.00	490.00	0.48	BZ-ZZ-22		28.97	107.17	90.95
230514	44714	23051	3.00	LM	0.00	76.00	228.00	0.22	BZ-ZZ-22		13.48	49.87	42.32

Fig. 9 Combined screenshots of tables with Movements, Road Trip, Shipments and Legs after the trip has finished

of distance-based and fuel-based footprints. Such a comparison is useful for two reasons: a) in order to get a better idea about the size of the accuracy gain of the fuel-based method, and b) in order to continuously assess the impact of the PCF calculations on the modification and accuracy of emission factors.

Generalizability of the reference architectures to other industrial sectors Of course each industry has its own specific processes, which limits the applicability of the proposed architecture. This means that for each industry a specific CMS reference architecture must be elaborated. Nonetheless, the methodology followed in this study for the logistic sector can be reproduced in other sectors as well. Furthermore, we believe that the main modules identified in the application layer should stay the same, although their concrete implementation may vary, per industry.

Appendix A

The CMS packages have been compared on several criteria that include all carbon footprint aspects discussed in this study and other offered functionality and technological aspects of the solutions. Concerning the functionality the comparison criteria are:

- Approach: does the vendor calculate footprints according to a certain protocol and emission factors, or are various ones supported? Is the IS restricted to GHG emissions, or can it also manage other polluting emissions? Does the package only provide functionality on GHG topics, or are other environmental management functions included, or at least available in external modules?
- Footprint calculation: can the CMS calculate both CCFs and PCFs, and are more specified footprints available?
- Monitoring: how are the footprints calculated and presented real-time?
- Reporting: what kind of reports can be produced? Does the CMS allow for auditing of the process?
- Acting: besides monitoring and reporting, it is valuable to act upon the footprint data. How is the data being analysed and what tools are available to the user to support reduction initiatives. How thoroughly are these actions planned and does the CMS keep track of the progress?
- Accounting: does the CMS offer functionality for accounting of energy and carbon offsets, and allowances trading?

Concerning the implementation, the CMS packages have been compared on:

- User management: can user roles or workflows be specified?
- Data entry: how can emission data be collected and entered into the IS?
- Integration: is the CMS able to communicate with other ISs and is integration with supply chain partners possible?
- Deployment: is the IS deployed locally, web-based or provided as a SaaS solution?

The table below shows an overview of the functionality and implementations characteristics of all surveyed packages (Table 2).

Table 2 Overview of CMS functionality

	Camco	CarbonView	Enablon	Enviance	Greenstone	Hara	IHS/ESS	Johnson	PE Int	SAP CI
Approach										
Protocol	GHG	++	++	++	++	++	++	GHG	++	GHG
Emission factors	IPCC		++	++	++	++			++	++
Scope of emissions	GHG	GHG	GHG+	GHG	GHG	GHG	GHG+	GHG	GHG+	GHG
GHG vs Eco	Eco	GHG	in/ex	in/ex	in/ex	Eco	in/ex	GHG	Eco	Eco
Footprint calculation										
CCF	v	v	v	v	v	v	v	v	v	v
PCF		v				v	v		v	v
Process		v			v	v	v		v	v
Departments	v	v	v	v	v	v	v	v	v	v
Distance calculation					v					
Change management			v				v			
Monitoring										
Dashboards	v	v	v	v	v	v	v	v		v
Static/live	Live	Live	Static	Static	Static	Live	Live	Static	Static	Live
Alerts		v		v						
Calendar				v						
Reporting										
Office formats	v	v	v	v	v	v	v	v	v	v
Compliance formats	CRC	v	v	v	v	v	v		v	v
Auditing		v	v	v	v	v	v		v	v
Sharing	v				v	v		v		
Acting										
Data analysis	Auto	Man	Man	Man	Man	Auto	Man	Man	Auto	Man
Benchmarking internal/external				Int	Int	Ext	Ext	Int		Int
Forecasting		v	v		v	v	v	v	v	
Scenarios			v		v	v	v		v	
Reduction planning		Man			Man	Auto	Man			Man
Targets	v	v	v	v	v	v	v		v	v
Financial impact	v	v			v	v	v			v
Risks		v				v				v
Progress		v		v	v	v	v		v	v
Accounting										
Energy	v		v				v			

Table 2 (continued)

	Camco	CarbonView	Enablon	Enviance	Greenstone	Hara	IHS/ESS	Johnson	PE Int	SAP CI
Offsets		v	v				v			v
Allowances		v	v	v			v			v
User management										
User roles	v	v		v	v		v		v	v
User workflows		v		v						
Data entry										
Manual	v	v	v	v	v	v	v	v	v	v
File upload		v		v	v	v	v	v	v	v
Automatic	v	v	v	v	v	v	v		v	v
Mobile devices	v			v			v			
Integration										
IS in general	v	v	v	v	v	v	v		v	v
TMS		v			v					
Energy meters	v					v				
Supply chain		v				v			v	v
Deployment										
SaaS		v		v	v	v	v			v
Web-based	v		v				v	v	v	
Local			v				v		v	

Appendix B

Acronym	Complete name
ABC	Activity Based Costing
AIS	Accounting Information System
APS	Advanced Planning System
BC	Board Computers
BPMN	Business Process Modeling Notation (www.bpmn.org)
CANbus	Controlled Area Network bus
CCF	Corporate Carbon footprint
CMS	Carbon Management Systems
DBMS	Data Base Management System
FMS	Fleet Management System
GHG	Greenhouse Gases
GPRS	General packet radio service (http://en.wikipedia.org/wiki/General_Packet_Radio_Service)
GPS	Global Positioning System
GSM	Global System for Mobile Communications (http://en.wikipedia.org/wiki/GSM)
IS	Information systems
LCMS	Logistics Carbon Management System
LSP	Logistic Service Providers
OES	Order Entry System
PCF	Product Carbon Footprint
SMS	Short Message Service (http://en.wikipedia.org/wiki/Short_Message_Service Message Service)
TMS	Transport Management Systems
TOGAF	The Open Group Architecture Framework (TOGAF 2009)
UML	Unified Modeling Language (www.uml.org)
UMTS	Universal Mobile Telecommunications System (http://en.wikipedia.org/wiki/Universal_mobile_telecommunications_system)
WMS	Warehouse Management System
XML	Extensible Markup Language (http://www.w3.org/XML/)
XSD	XML Schema Definition (http://www.w3.org/XML/Schema.html)

References

Ageron, B., Gunasekaran, A., Spalanzani, A. (2011). Sustainable supply management: an empirical study. *International Journal of Production Economics*. doi:10.1016/j.ijpe.2011.04.007, Available online 19 April 2011, In Press, Corrected Proof.

Bass, L., Clements, P., & Kazman, R. (2003). *Software architecture in practice* (2nd ed.). Reading: Addison-Wesley.

Boer, L., Brouwer, F., & Essen, H. v. (2008). *STREAM Studie naar TRansport Emissies van Alle Modaliteiten*. Delft: CE.

Boulter, P., & McCrae, I. (2007). *ARTEMIS: Assessment and reliability of transport emission models and inventory systems: final report*. Wokingham: TRL report.

BSI. (2008). *PAS 2050:2008 Specification for the assessment of the life cycle greenhouse gas emissions of goods and services*. London: Bristish Standards Institute.

CapGemini (2007). Transportation report 2007, retrieved January 18, 2011, from:http://www.capgemini.com/insights-and-resources/by-publication/transportation-report-2007/.

Chapman, L. (2007). Transport and climate change: a review. *Journal of Transport Geography, 15*, 354–367.

Chen, A. J., Boudreau, M., & Watson, R. T. (2008). Information systems and ecological sustainability. *Journal of Systems and Information Technology, Sustainability and Information Systems, 10*(3), 186–201.

Chen, A. J., Watson, R. T., Boudreau, M.-C., Karahanna, E. (2009). Organizational adoption of Green IS & IT: An institutional perspective. *ICIS 2009 Proceedings* (p. Paper 142).

EEA. (2009). *EMEP/EEA air pollutant emission inventory guidebook 2009*. Copenhagen: European Environment Agency.

European Commission (2008). *Climate change: Commission welcomes final adoption of Europe's climate and energy package.* Retrieved May 3, 2010, from Europa Press Releases RAPID: http://europa.eu/rapid/pressReleasesAction.do?reference=IP/08/1998.

Finkbeiner, M. (2009). Carbon footprinting—opportunities and threats. *The International Journal of Life Cycle Assessment, 14*(2), 91–94.

Greefhorst, D., Grefen, P., Saamen, E., Bergman, P., Van Beek, W. (2009). Herbruikbare architectuur - een definitie van referentie-architectuur. *Informatie, September 2009*, 8–14 (in Dutch).

Gregor, S., & Jones, D. (2007). Anatomy of a design theory. *JAIS, 8*(5), 312–335.

Groom Energy Solutions (2010). *Enterprise carbon accounting.* Retrieved from http://www.groomenergy.com/eca_report_summary.html.

Hevner, A. R., March, S. T., & Park, J. (2004). Design research in information systems research. *MIS Quarterly, 28*(1), 75–105.

Holtkamp, B., Steinbuss, S., Gsell, H., Loeffeler, T., Springer, U. (2010). Towards a logistics cloud. In *Proceedings of 2010 Sixth International Conference on Semantics, Knowledge and Grids, 1–3 Nov.* (pp. 306–308).

Iacob, M. E., Jonkers, H., Lankhorst, M., & Proper, H. (2011). *ArchiMate 2.0 specification*. Zaltbommel: Van Haren Publishing.

IEA. (2009). *CO2 emissions from fuel combustion—2009 edition—highlights*. Paris: International Energy Agency.

IPCC. (2007). In Core Writing Team, R. K. Pachauri, & A. Reisinger (Eds.), *Climate change 2007: Synthesis report. Contribution of Working Groups I, II and III to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*. Geneva: IPCC.

Kuo, B. N., & Dick, G. N. (2010). Organizational Green IT: It seems the bottom line rules. *AMCIS 2010 Proceedings* (p. Paper 99).

Mann, H., Grant, G., Singh Mann, I. J. (2009). Green IT: An implementation framework. *AMCIS 2009 Proceedings, Paper 121.*

Matthews, H. H. (2008). The importance of carbon footprint estimation boundaries. *Environmental Science and Technology, 16*(42), 5839–5842.

McKinnon, A. (2010). Product-level carbon auditing of supply chains: environmental imperative or wasteful distraction? *International Journal of Physical Distribution and Logistics Management, 40* (1–2), 42–60.

Mendix (2011). *Mendix reference guide 2.5*. Retrieved from: https://world.mendix.com/display/NRG.

Molla, A. (2008). GITAM: A model for the adoption of Green IT. *ACIS 2008 Proceedings* (p. 64).

Naumann, S., Dick, M., Kern, E., & Johann, T. (2011). The GREENSOFT Model: a reference model for green and sustainable software and its engineering. *Sustainable Computing: Inforatics and Systems, 1*(4), 294–304.

Ntziachristos, L. G. (2009). COPERT: A European Road Transport Emission Inventory Model. In I. M. Athanasiadis (Ed.), *Information technologies in environmental engineering* (pp. 491–504). Springer.

Peffers, K., Tuunanen, T., Rothenberger, M., & Chatterjee, S. (2008). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45–77.

Rich, D. (2008). *New Greenhouse gas protocol standards.* Retrieved May 31, 2010, from International SmartWay Transportation Summit: http://epa.gov/smartway/transport/documents/international/david-rich-world-resourcces-institute.pdf.

TLN (2008). *TLN Infoblad nummer 143. Resultaten TLN Automatiseringsenquête 2008.* Transport en Logistiek Nederland (in Dutch).

TOGAF (2009). *The open group architecture framework, version 9, enterprise edition. Document number: G091.* Retrieved November 29, 2010, from The Open Group: http://www.opengroup.org/architecture/togaf9-doc/arch/.

Verdantix (2009). *Green quadrant: Carbon management software (Global)*. Retrieved from http://www.verdantix.com/index.cfm/papers/Products.Details/product_id/51/green-quadrant-carbon-management-software-global-/-.

W3C (2004). *Web services glossary.* Retrieved June 22, 2010, from http://www.w3.org/TR/ws-gloss/.

Wackernagel, M., & Rees, W. (1996). *Our ecological footprint—reducing human impact on the earth*. Gabriola Island: New Society Publishers.

Walls, J., Widmeyer, G., & El Sawy, O. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research, 3*(1), 36–59.

Watson, R. T., Boudreau, M.-C., Chen, A., & Huber, M. H. (2008). Green IS: Building sustainable business practices. In R. T. Watson (Ed.), *Information systems*. Athens: Global Text Project.

Webb, M. (2008). *Smart 2020: Enabling the low carbon economy in the information age.* http://www.theclimategroup.org.

Weidema, B. T. (2008). Carbon footprint—a catalyst for life cycle assessment? *Journal of Industrial Ecology, 12*(1), 3–6.

Wiedmann, T. (2009). Carbon footprint and input–output analysis—an introduction. *Economic Systems Research, 21*(3), 175–186.

Wiedmann, T., & Minx, J. (2008). A definition of 'Carbon Footprint'. In C. C. Pertsova (Ed.), *Ecological economics research trends* (pp. 1–11). Hauppauge: Nova.

Zadek, H., & Schulz, R. (2010). Methods for the calculation of CO2 emissions in logistics activities. *Advanced manufacturing and sustainable logistics. Lecture notes in business information processing, 2010, volume 46, part 3* (pp. 263–268). Springer.

Maria-Eugenia Iacob is currently associate professor in the Industrial Engineering and Business Information Systems Department at the University of Twente. She holds a Ph.D. in Mathematics from the University Babes-Bolyai of Cluj-Napoca (Romania). She is lecturing and doing research in the areas of enterprise systems architectures, service-oriented architectures, model-driven development, business process (re)engineering, business rules, and information systems. She is one of the developers of ArchiMate and co-author of the ArchiMate 1.0 and 2.0 standard specification. She has published more than 50 scientific papers and books.

Marten J. van Sinderen is Associate Professor at the University of Twente, The Netherlands. There he is also leader of the strategic research orientation "Applied Science of Services" of the Centre for Telematics and Information Technology. His major research interests

are design methods and architectures for networked information systems, particularly in the area of next-generation collaborative enterprises and smart consumer applications. He participated and managed several research projects in the aforementioned area and co-authored more than 100 research papers. He was General Co-chair of EDOC'05, Program Co-chair of ICE-B'07-'12, IWEI'08-'12 and ICSOFT'12, and Steering Committee Chair of EDOC'11-'12.

Michel Steenwijk holds a master degree in Business and IT from the University of Twente. He has carried out his graduation project at CAPE Groep. In this capacity, he was involved in the project Activity Based Carboning (ABCO2), and he participated in the implementation and design of a carbon management system.

Pieter Verkroost is a Senior Consultant at CAPE Groep. In this capacity, he is involved in the company with project management, consultancy, agile business modelling and special projects. He and CAPE Groep deliver model-driven software solutions with a service-oriented architecture, mainly in the area of supply chain processes specialized in transport and logistics. He was the project manager of the project Activity Based Carboning (ABCO2). In this project the goal was to calculate and simulate CO2 per order, based on a pre-defined multi modal transport network with normative emission factors.