Intelligent Infrastructure for Last-mile and Short-distance Freight Transportation with Electric Vehicles in the Domain of Smart City Logistic

Volkmar Schau, Sebastian Apel, Kai Gebhardt, Johannes Kretzschmar, Christian Stolcis, Marianne Mauch and Johan Buchholz

Department of Computer Science, Friedrich Schiller University Jena, Ernst Abbe Platz 2, Jena, Germany

Keywords: Smart City Logistic, Electric Vehicle, Software Architecture, Range Prediction, Logistic Ontology, Acceptance Study, Usability, Simulation, Generic Interface, Intelligent System.

Abstract: The current living standard of industrial nations causes increasing CO2 emissions, particulate matter, and noise pollution. An essential amount of these environmental issues is induced by stop-and-go traffic within cities which is seriously characterized by short-distance freight transport trips with inner-city and suburban distances. The Smart City Logistik project strives for a practical and short-term solution to this problem in the concrete context set by the city of Erfurt, Germany. This paper provides the results of an open and intelligent infrastructure for transportation with electric vehicles. The special focus is on holistic reflection of ICT support in the electric city logistic.

1 INTRODUCTION

The launch of the German national development plan for electro mobility (Bundesregierung Deutschland, 2012) has spawned some activities and projects, ranging from research to industrial development, and sporting goals with short-term as well as long-term lifelines. In most cases the obvious shortcomings of currently available, fully electric vehicles (EVs) are addressed and tackled with a particular mix of various technologies.

The main problem identified was, of course, the limited range of available cars. Long-term projects focus, in this context, to a large extends on the development of innovative battery technologies. However, most researchers agree that until 2020, and well beyond, batteries will not be able to guarantee driving ranges close to what can be achieved today with traditional gasoline-driven engines, as least if a viable weight to power ratio must be the goal. Thus, a number of alternative projects have taken as a premise that we will have to cope with limited ranges for at least the next decade. Based on this assumption the challenge is to support available EVs by other technologies to reach maximum usability. One of these alternatives, and maybe the most important one, is information and communication technologies (ICT).

The special federal research program Information and Communication Technologies for Electric Mobility II (ICT II) has, thus, been established in Germany to leverage the capabilities of ICT-based research by adapting available and new concepts to the individual needs of electro mobility (Bundesregierung Deutschland, 2013). The projects funded by this program strive mostly for short and medium-term solutions with a clear focus on immediate applicability and an early market entry. The coordinator and manager of this research program is the German Federal Ministry for Economic Affairs and Energy (BMWi).

Achieving these goals, is done by bundling institutions with industrial partners within related service domains. Thus, each research project combines EV suppliers, end-users (application partners, companies, and individual drivers), and other domain-specific players (in this case a provider for logistics and fleet management software). The so formed consortium will not only strive to reach some well-structured research goals but will work towards a prototype solution industrial strength. Thus, besides research and development objectives the project will also address questions regarding marketing and sales. Deliverables include not only papers and concepts but, as already mentioned, readily available tools and a viable evaluation of the integrated solution in a practical setting

Schau, V., Apel, S., Gebhardt, K., Kretzschmar, J., Stolcis, C., Mauch, M. and Buchholz, J.
Intelligent Infrastructure for Last-mile and Short-distance Freight Transportation with Electric Vehicles in the Domain of Smart City Logistic.
In *Proceedings of the International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2016)*, pages 149-159
ISBN: 978-989-758-185-4

(major field test). In most cases this leads to a consortium that has a well-established track record in a particular application domain. Also, a real life test-bed has to be located to enable the physical field test.

The Smart City Logistik (SCL) project (DAKO, 2015) targets the application domain of inner city merchandise traffic. The concept is to unload cargo from heavy trucks on the citys perimeter and to run the last miles with small and medium sized EVs. In most cases the logistics partners also utilize storage facilities outside the city to provide additional buffer capabilities and to decouple long-range from short-range traffic in this way. The city of Erfurt, in Thuringia, Germany, has been chosen as test-bed because it has passed stringent laws regarding inner city transport that will, most likely, favor EVs as the main transport medium of the future.

The challenge is to support this fleet of transport-EVs with an ICT system that provides for an integrated interface to existing logistics systems, as well as to estimate and manage each individual vehicles range, itinerary, and routes with a highly adaptive solution. Based on limited battery capacity, always changing environmental conditions, the usual short-term necessary adaptations in the planned itinerary, and stringent legal requirements SCL expects a steady rate of exceptions that have to be handled by the system in real-time. Besides, there will be no guarantee that all EVs in the fleet will be online all the time, simply due to possible problems with the cell-phone network in more remote areas, during inclement weather, or because of overload and technical defects.

From a technological point of view (see figure 1) the SCL system is a distributed, mobile ICT system with a central server unit and fat driver assistant client (DAC) that have the capability to run independently from the main server while computing, at least, all essential services in the case of a disconnect. The central server itself has to be usable as a standalone service as well as a system in addition to existing system landscapes within logistical companies. Runtime consumption and vital data from EVs in this setup have to be collected and transmitted by using telematic units as interfaces for EVs controller area network (CAN) busses. Furthermore, it is important to know about the acceptance of drivers by using the DAC. By performing multiple acceptance tests within SCL, not only by testing the DAC, it is possible to ensure the usability of the resulting systems. As a federally funded project, SCL also strives for a solution that is not proprietary to a company, open for changes and additions in the future, and based on a widely published and standardized ICT-architecture (Schau et al., 2015).

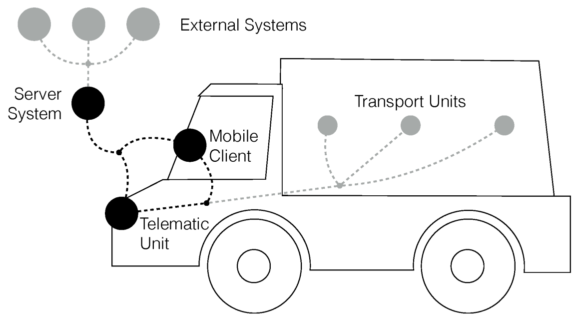


Figure 1: SCL platform overview: in-car elements like DAC, transportation units and telematic unit communicates over the air with a centralized server which is used as a planing, monitoring and data system and can be used as a bridge through external systems.

2 RELATED WORK

The German BMWi funds multiple projects which investigates into the key subjects of ICT II. The focus of ICT II is on new concepts for intelligent technology in EVs (Smart Car) combined with power supply (Smart Grid) and concepts for mobility (Smart Traffic) (Bundesregierung Deutschland, 2013).

The ICT II funded collaborative research project sMobility tried to link existing components of the infrastructure using an open ICT-platform. Special focus is on price controlled and decentralised charging of EVs and on navigation, which is optimized for journey time using actual traffic and car data, as well as on intelligent flow regulation technology. In the city of Erfurt, an intelligent and linked transport infrastructure is constructed. Related live data is provided within a distributed cloud system (INNOMAN, 2015). Like SCL sMobility created a range prediction model to optimize the route considering local traffic information. A unique characteristic of sMobility is the collection of traffic data using detectors along the road. The sMobility range prediction model is less specific than SCL and focusing on intermodal and private transportation. On the other hand, the high-detailed data can be used within SCL as possible data sources for more precise range predictions.

iZEUS, which was also funded by ICT II, considered more complex standards for controlled charging of EVs. During the whole day, we have big fluctuations in the power supply because of regenerative energy like solar and wind produces. Therefore, the project tried to regulate the charging intensity in a way the drivers needs won't be restricted. Decentral storage in EVs with an energetic recovery system is one of their primary project focus. An onboard charging system and an efficient energy management with an infrastructure for communication between two actors of a B2B car fleet was conceptualized. A service detects the next electric charging station and municipality. Private and business customer tested this technology (EnBW, 2015) and evaluates their usability within an acceptance survey. In contrast, to SCL the focus is on the calculation of efficient routes and visualization of remaining ranges to specific destinations. Unfortunately, the management and range prediction isn't working together as required in SCL to support an installation within most logistic scenarios.

Adaptive City Mobility (ACM), funded by ICT II, develops an electric micro car with a portable system to change rechargeable batteries for urban traffic and logistic - used as eTaxi fleet (VISPIRON CARSYNC, 2015). As mentioned in other projects before ACM linkd several components like the ICT of the EV, mobile units, central cloud services and - unique for this project - charging stations. Concerning of self-designed electric micro car and their telematic unit ACM can provide information about the state of the rechargeable battery, the remaining range and the next charging station within their mobile client.

eTelematik was a second collaborative research project in cooperation with Friedrich-Schiller-University Jena (Navimatix, 2015) and was funded by Zentrales Innovationsprogramm Mittelstand (ZIM). The project focused on a solution for municipality using a special prototype of Multicar as a EV. That EV can be utilized (e.g.) as a road sweeper or a snow plow through different extensions. The Multicar was equipped with an individual telematic unit, to collect data, and with a driver assistance system. The realized system platform focused on task management, and the enclosed range estimation model is based on neuronal networks. This kind of model leads to machine learnable results but is expensive in calculation (Beikirch, 2015).

E-Wald, funded by a Bavarian research program, is a collaborative research project in Bavarian Forest that aims at rural areas. Using newly developed and intelligent charging devices and communication concepts E-Wald wants to proof that electric mobility works in that rural areas as well. The project is developing a new generation of a fast charging station for all charging systems. Focus is on a wire- and plugless charging concept. Just as SCL it uses a range estimation model. The range estimation model uses geographical information, the state of the rechargeable battery, outdoor temperature, driveability, as well as information about topography and road system based on a statistical model. The range estimation, used by a mobile client, will be shown like a blue sky on the map and will be calculated car dependent (Technische Hochschule Deggendorf, 2015). Step by step E-Wald optimized their range estimation model by adding one parameter at the time. SCL examined the parameter which influences the range of an EV at first. The range estimation model was designed afterward.

3 ARCHITECTURE

SCL combines highly frequented telematic data, central server infrastructure and a DAC to get EV more useable, safe in planning and perfectly usage of their ranges. Starting with interviewing dispatchers and drivers about their everyday work in inner-city logistic companies (Apel et al., 2015a) shows a brief insight into their existing processes and used system landscape. Logistic companies mostly use specialized supply chain management (SCM) systems like transport management systems (TMSs) or warehouse management systems (WMSs) to manage their orders. The decision, which system is used, depends on their business orientation. Companies with a focus on selling products would use WMS to manage their stocks and orders in addition to transportation planning. Conversely, companies with a focus on transportation and route management will use TMS to manage their available transportation resources. Unfortunately, these systems don't care about particular range restriction of EVs and their requirements for transportation tours. Handling this range restrictions and requirements are, were the SCL platform has to be embedded.

The SCL platform wants to provide planning and monitoring tools for EVs which has to be used in addition to existing processes and system landscapes in logistic companies. To get in this existing system landscapes, two analyses has to be done: what's about their processes and which software is involved, more specifically which interfaces can be used to get related data. Processes could be analysed and generalized by using the previously done interviews (Apel et al., 2015b) and validated in literature about SCM (Chopra and Meindl, 2007) and logistic processes (Arnold et al., 2002). The last points, about interfaces, leads to a difficult challenge within this domain. Taking a closer look at available systems shows a heterogenic interface landscape (Busch et al., 2003; Logistik Heute, 2005; Pirron et al., 1999; Faber and Ammerschuber, 2007). There are several process standardizations like RosettaNet[1] and United Nations Electronic Data Interchange for Administration,

[1]RosettaNet is globally organized between nearly 600 companies and managed by the GSI US, defines logistical processes and terminology.

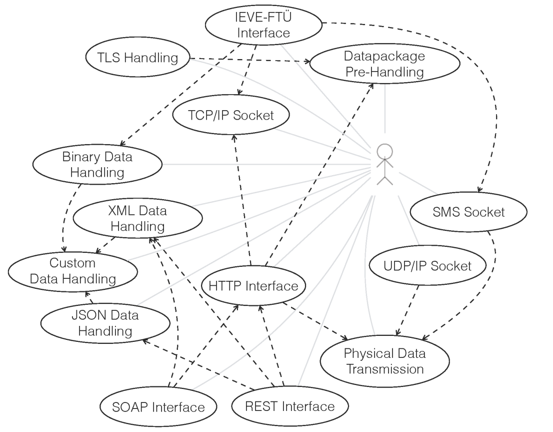


Figure 2: Use cases for interfaces within the SCL system landscape.

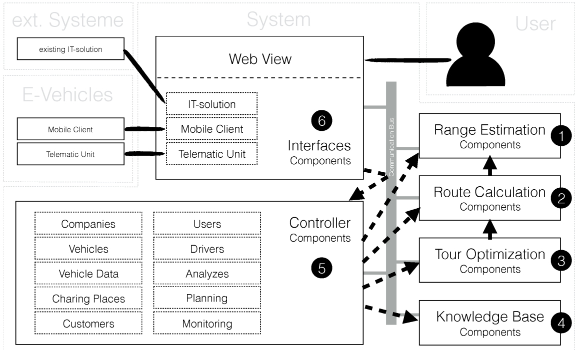


Figure 3: Overview of the SCL architecture. Each model, controller and view component interacts within this system by using a communication bus for private and broadcast messages.

Commerce and Transport (UN/EDIFACT)[2] influencing the way these systems work, as well as what terminology to use within this domain. Unfortunately, the data exchange itself is not standardized. There is a broad range of proprietary, Representational State Transfer (REST) or Simple Object Access Protocol (SOAP) interfaces with Extensible Markup Language (XML), JavaScript Object Notation (JSON) or again something proprietary, as well as electronic data interchange (EDI) based file exchange or custom American Standard Code II (ASCII) file export mechanics. The most common interface can be found for financial information, like DATEV, sadly this one would not help at all. Adding those known interface use cases in addition to required interfaces to the DACs and the telematic units would lead to a wide range as shown in figure 2.

To achieve these goals several required components has to be orchestrated within an architecture as shown in figure 3 for planing, monitoring and analyzation. The first one is the range estimation component itself (1). This component should have capabilities to handle precise forecasts on planed routes, as well as mechanics to learn from already driven routes. After that a route calculation component (2) can use this range component to score routes, and in turn can be used from tour optimization component (3) which combines different routes in one tour to get an energy optimized result. Data within this system, in detail every related entity like drivers, vehicles, orders and shipments, as well as charging stations, users, companies and customers, has to be handled and get associated in a knowledge base component (4). Beside this a bunch of components to handle different types of requests has to be added (5). Those components handles requests about already named tasks like monitoring, planning and analyzing, as well as managing their related core data like available vehicles, drivers, users and customers in addition to basic data like vehicle type data[3] and charging places. All components has to communicate by using a internal communication bus. Components can listen for events, release event and start a private communication with an other component. This part is based on an observer pattern (Gamma et al., 1994, p. 326), futhermore those observers would't be registered for special observables, they will be registered in a global context, which creates something like a enterprise service bus (Chappell, 2004, p. 2) between those components. External systems can be embedded within this architecture by adding a specific interface component (6) which transformes external communication flows to internal ones. This architecture draft with their model components (range, route, tour and knowledge), controller components and interface components adaptes the Model-View-Controller (MVC) (Buschmann et al., 1996, p. 125) pattern to minimize coupling and maximize cohesion.

Challenging the problems with this broad range of possible external systems is done by splitting up each part of a communication between two participants. In preparation three strategies were developed to handle

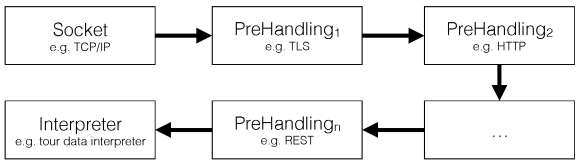


Figure 4: Pipeline strategy for interface components.

[2]Known as X12 from USA and EDIFACT from Europe and combined to the global standard UN/EDIFACT can be used as exchange standards for logistic related data.

[3]Each vehicle instance depends on a previously defined vehicle data entry.

this communication in SCL. The first one handles this broad range by creating a inheritance tree for different specializations of interfaces. The second, finally used within this architecture, made use of the pipeline concepts (Buschmann et al., 1996, p. 53) to split each transformation step within a communication stack defined by the OSI reference model (ISO/IEC JTC 1, 1994). Each step like Transmisson Control Protocol (TCP), Hypertext Transfer Protocol (HTTP), Transport Layer Security (TLS) can be rearranged in a fixed order to handle a specific communication stack like shown in figure 4. The last one describes a strategy where the interface itself isn't handled directly. This means, the architecture describes well defined application programming interfaces (APIs) for each existing request type. An interface developer has to use this API and implement the whole interface as a plugin for this system. Because of the expanding number of dependencies in case of using inheritance trees and heavily not reusable characteristic of an API strategy, the SCL platform uses the pipeline strategy to get configurable interfaces for wide ranges of existing systems.

The system described above and realized within SCL can be used as a basic infrastructure within the SCL platform. The next step would be to embed the corresponding components like range estimation and knowledge base. Furthermore, the entities defined by this knowledge base can be used to describe a data exchange model for component interactions within this internal communication bus.

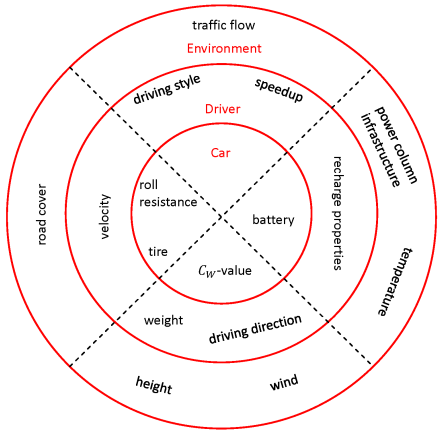


Figure 5: Influence factors on energy consumption by driving an EV (Conradi, 2012).

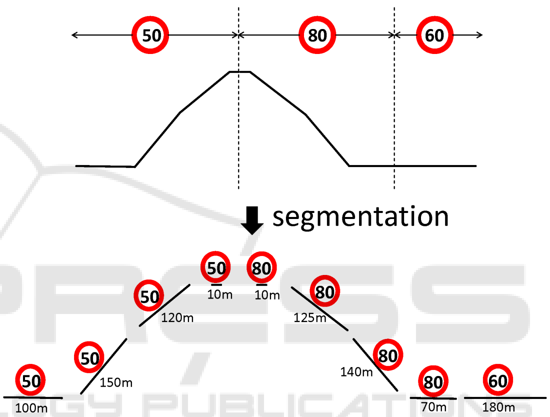


Figure 6: Segmentation of a tour.

4 RANGE ESTIMATION MODEL

The use of EVs in the transport sector is challenging because the range of available small and medium sized transport vehicles is relatively small. To handle this challenging usage, a precise range estimation method is needed, which can calculate the energy consumption of a tour as closely as possible. This estimation method is a difficult task, because many factors have an effect on the energy consumption of an EV. Significant influence factors are shown in figure 5 (Conradi, 2012).

So a research goal of this work is to provide such a range estimation method. A weakness of existing solutions is a very high dependency on the regarded car type. Rogge et al. tries to model a Mitsubishi i-MiEV in Matlab and uses as influence factors the car's weight, average energy consumption of an i-MiEV, ramp of road, air resistance and frictions. Most of this parameters are vehicle specific. Ondruska and Posner use a similar approach with a combination of stochastic methods and physical considerations (Ondruska and Posner, 2014). This approach bases on technical aspects of a Nissan Leaf.

Therefore, an algorithm without physical considerations can avoid these disadvantages. The energy consumption of a new tour is predicted by comparing the tour with known tracks. In order to perform this task, a similarity measure between tours is necessary. First, a model is developed to describe a tour by dividing it into segments. Figure 6 shows an example. A tour is analyzed from start to end and a new segment begins at each point where at least one parameter changes the value. The example uses the parameters velocity and ramp of road. It follows that a segment can be described by constant values in each influence factor. As presented in figure 7, groups are formed by using the cross product of segments for each influence factor. Each group is visualized with one cuboid. For example, velocity is divided into three intervals with the limits zero to ten, ten to twenty and twenty to thirty.

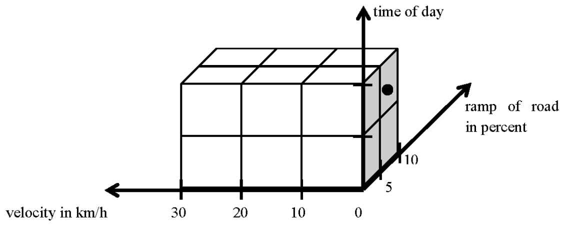


Figure 7: Visualization of groups.

The estimation method is based on a learning data set. Which use segments, with a known energy consumption value. The telematic unit used in SCL generates those consumption values in the case of tour execution. Using these measured values as learning data set can update the range estimation model without additional development efforts. The learning data segments are sorted into groups depending on their values for the influence factors. Then for each group the average energy consumption of segments in this group is calculated. On this way, a look-up table is generated containing the energy consumption of each group.

To estimate the consumption of a new segment, the known factors of influence of this segment has to be used to get the related group. The average energy consumption of this group is used as an approximation of the energy consumption for this new segment. The approach was tested on a real data set, which was generating with a Mitsubishi i-MiEV. The result shows that this prediction model has a relative error about eight percent. This relative error is a good standing similar to the approach of Ondruska and Posner. A benefit of this approach is the run-time. New segments can be predicted by looking into the pre generated look-up table. The estimation method is described in more detail in (Gebhardt et al., 2015).

5 KNOWLEDGE BASE

Concerning software architecture for supporting EVs in logistics as introduced in section 3, a comprehensive conceptual data model had to be developed. This model should be able to support the delegation and monitoring of transport chains and tasks in all kind of versatile logistic scenarios. Due to the requirements analysis, there were several problems:

- The data model should be easily and potentially semi-automatically adoptable to all kinds of different logistic scenarios. The three application scenarios of SCL serve here as a benchmark.
- Data aggregation and management is supposed to be directly supported by the data model itself. This point is especially targeted to the fact, that the data model should include some kind of semantic data layer. By this, it would be possible to combine and integrate databases of different application partners.
- It should enable gaining extensive implicit information and data, which support the disposition of tours, staff and resources actively.
- Finally, the data model should cover especially the electric specific aspects of logistic domains.

The generic requirement and the extraction of implicit information suggested the use of a descriptive, logic-based specification for the data model. There are far reaching works in this field of research in the context of logistic applications like (Lian et al., 2007) or (Jinbing et al., 2008). The SCL approach is based on existing formal ontologies like GenCLOn (Anand et al., 2012), which provide an essential vocabulary to describe logistic concepts and relationships. The use of quasi-standards also enables the subsequent implementation and application of the SCL system by referencing to top domain models.

Unfortunately, none of these existing ontologies and models covered all necessary aspects of the underlying SCL application area. Therefore, the conceptual model has to be extended due to the primary use of EVs at first. These enhancements were made especially regarding the influence parameters for the range of EVs as described in section 4. The limited range and charging behaviors are a critical factor for the acceptance of application of EVs and should be therefore supported by the data model. The goal hereby is to assist the dispatcher and the DAC to identify early flaws of a given tour or execution state.

Furthermore, the SCL requirement analysis showed, that the existing ontologies only provide a very general and common vocabulary to include concepts and relations of a logistics domain. So besides the electric driven aspects, these ontologies had to be extended by definitions to enable the description of legal regulations like driving times, hazardous cargo or driving licenses classes. These concepts are likewise very heterogeneous and difficult to recognize, especially in transnational contexts but indispensable for supporting the dispatching of logistic tours. The SCL system is supposed to be a comprehensible assisting system, which is able to recognize legally invalid tours by checking the consistency of a given rule base. Although such a rule base provides a very elegant way to extend and adapt to future use cases or potentially changes according to the law without the need for changing data processing algorithms based on a underlying model. This approach is described on the basis of driving licenses according to the EU

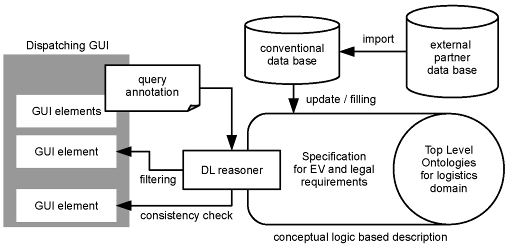


Figure 8: Integration of the knowledge base as semantic representation of data models and assistance for dispatching and monitoring logistics.

norms in (Prinz et al., 2015) in detail. Here the implementation of the rule base occurred in a Prolog program and information was retrieved by reasoning. By this the SCL system is able to support the dispatching of logistic tours as systematic queries restricted given options. Errors could be prevented by early consistency checks and even corrected automatically.

The ontology, respectively the rule base, represents the formal semantic layer of a given data model, which is provided by an external application partner. Therefore, there has to be a specific import algorithm for every data model, which realizes the inflation of the ontology. By this, the ontology only serves as an inferential database. Database changes are made in the common data model and are imported back to the semantic layer. Due to the complexity of logic descriptions and inferential data, it is not feasible to transfer data from the semantic description to conventional relational databases.

Furthermore, not every end user, especially in the field of logistics dispatching, can be assumed to be a knowledge representation engineer. So, there has to be some kind of opaque access and inference mechanism. This problem is faced by combining a conventional graphical user interface (GUI), like it is used for tour dispatching, with the ontology reasoner. Every element of the GUI gets annotated with a specific query, which gets invoked as soon as some GUI elements get changed, respectively there is input data available. So, if the dispatcher enters data in one GUI field, the reasoner handles all queries tries to infer data for all the other elements of the GUI. By this, the options for one element can be restricted or even incongruities derived if there is no valid option anymore available.

In summary, the knowledge base is integrated into three main aspects within the SCL architecture, as shown in figure 8. These aspects cover all the exposed requirements from the beginning of this section. First, the knowledge base represents a specification of given top-level domains in the field of logistics. The SCL ontology especially describes electric driven aspects as well as further legal dependencies. Secondly, the knowledge base typifies a semantic layer for given external conventional data models. And lastly, there is a tight connection between a dispatching GUI, respectively the elements of such a GUI, and ontology queries.

6 DRIVER ASSISTANT CLIENT

One of the most important parts in such a holistic system is the assistance of the driver. In a complex socio-technical system where EVs are used for inner-city logistic, the driver needs to cope with additional information (such as range, battery status, etc.) in order to fulfil the main task of delivering goods. To reduce stress and uncertainty resulting from these additional important parameters, a DAC was developed and implemented as a prototype during the SCL project. Basically, the DAC works as a navigation system, providing optimized routes for EVs used in the project, considering different range-affecting parameters of the EV (see chapter 4) while focussing on the planned tour.

Since the DAC should assist the driver while driving and reduce the added complexity, it needs to be easy-to-use in a vehicle. To find an optimal way to support the driver two different horizontal prototypes were developed at an early stage of the project.

Besides interviews with drivers and dispatchers, we systematically evaluated the user acceptance and usability of the DAC.

The main challenge was to meet with the participants in selected companies because of restrictions caused by standardized workflows and internal procedures. During the week there was almost no time to interview more than a few drivers per day, that made personal interviews or a supervised experimental setting inefficiently and costly. The alternative, an unsupervised setting for testing the DAC and filling out the questionnaire raises methodological challenges (e.g. how to guarantee a similar experimental set-up for each participant). To address this problem, a contact person from the participating companies was provided with detailed instructions about the experimental setting and a short 20-minute online survey was elaborated. Several methods like access codes, randomization and the analysis of metadata (e.g. time-stamps and IP-addresses) that was stored in our online-questionnaire, helped to address methodological concerns regarding the unsupervised setting.

In the beginning, each participant received an impersonalized envelope with information about the procedure and a unique access code from the contact

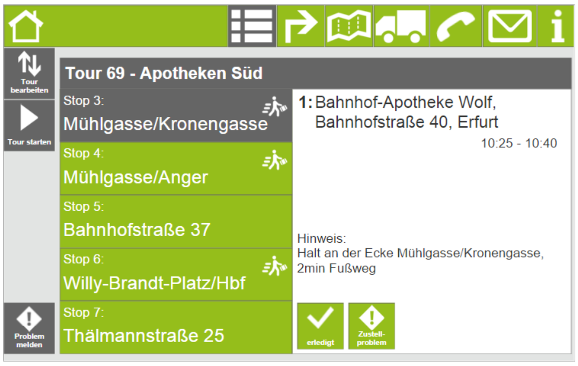


Figure 9: GUI prototype 1.

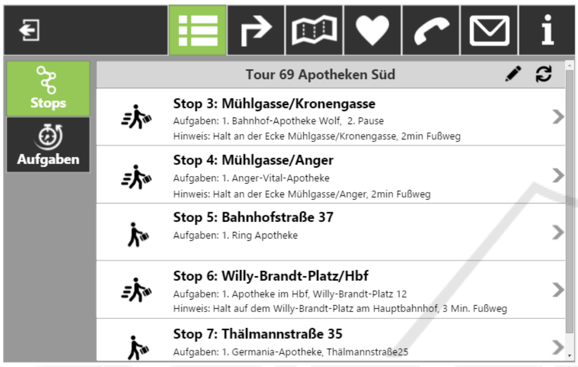


Figure 10: GUI prototype 2.

person in their company. Each participant had a personal computer with the online survey and a tablet showing the prototypes in front of him. The access code was needed to start the survey on the PC and the first prototype (which was displayed in randomized order) on the tablet. After answering some questions concerning general and specific attitudes towards EV and technological solutions in general, the participant was asked to perform basic tasks using the prototype. Based on this tasks, the participant had to answer some questions regarding the functionality and the design of the prototype of the DAC. Finally, the participant had to perform exactly the same steps also for the second prototype, answering the same questions as for the first one.

Except for general attitudes to EV and technology at the beginning and demographic attributes at the end, the questionnaire itself was created based on an operationalization of the ISONORM 9241-110 standard (Prümper, 2010; Prümper, 1993) for software evaluation, which has been adapted to the specific research context. For each prototype, the participants were asked to evaluate the adequateness, handling and intuitiveness. The questions itself could be rated on an ordinal scale with seven possible values between "- - -" and "+ + +".

While analysing the dataset, the access code was

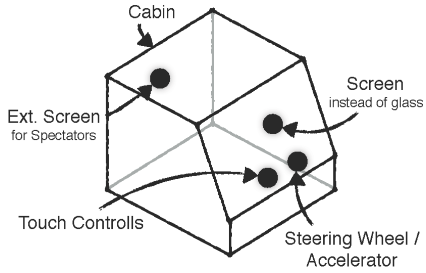


Figure 11: Eltrilo cabin schema.

used to assign the answers to the specifically evaluated prototype and to compare the answers on a company-level without revealing the participants identity. This procedure prevents a bias resulting from the order of prototype evaluation. Each access code could be used only one time. Additional security was added through an encrypted connection to the servers. Furthermore, the use of the online survey system allows to exclude incomplete or corrupted answers based on the time of completion of each segment and the participants IP-addresses.

Overall participating companies 43 participants filled out the survey evaluating the two prototypes, from which five could not be used because they were not completed. Regarding the two prototypes, it turned out, that none of them was evaluated better than the other. Prototype 1 as shown in figure 9 obtained better results regarding the usability while prototype 2 as shown in figure 10 achieved better results in clearness of the design and intuitiveness. Based on this results a set of adaptation recommendations was elaborated which basically combined the positive characteristics of both prototypes. Finally, this set of adaptations has been used, to build the fully functional prototype.

7 ELTRILO

Eltrilo is a cabin based on a former multi-purpose vehicle Multicar, which is famous in city and pedestrian areas. The cabin is fully featured with all equipment for the everyday work in courier, express and parcel area. But Eltrilo is not able to move. The cabin contains a simulated SCL environment in which a driver can process several orders. So in the background the SCL architecture, range estimation model and knowledge base is working. In the foreground the driver interacts only with the DAC by driving in a simulated city based on true roads with real-time position data.

Figure 11 presents the cabin schema. In front of the driver, a huge flat screen is the new virtual glass windshield. Besides the steering wheel, there are two additional small screens which form the cockpit. The cockpit touch screen allows to switch cabin equipment like radio, air condition, lights et cetera on or off. On the second screen, a navigation system and the cabin state like velocity is accessible. The driver assistance client is placed on top of the cockpit.

Entering the cabin the considered test drive is an express order from a suburb to the city center delivering ice cream. Time for the order is around 5 minutes. In the beginning, there is an area to be familiar with the EV. All the time a virtual assistant driver is helping per voice. In the test drive, there is an average working day and no rush hour. On midway, the weather pattern is changing. At the end of the trip, the driver gets a report how to improve their next tour.

Every 500ms Eltrilo's data logger writes a record of position, energy consumption and four additional parameters. Overall there are 10,822 trips with about 500 trip segments. So the entire data set contains 5,338,084 segments.

Processing the data set all segments with velocity zero are removed, and trip segment's energy consumption is normalized by distance. So the distance parameter could be eliminated in the range estimation model. Therefore, it is necessary to remove all items with a distance less than 0.001. Otherwise, energy consumption scores absurd high. As a result, 5,097,588 segments remain.

The next data processing step is to eliminate high-performance shifts in negative acceleration and energy consumption. Such shifts are produced by high-velocity crashes hitting a snag. This step is done by boxplot. That means for energy consumption items 89.4 percent are inside the interval [-19, 3, 32, 1] and for acceleration items 90,1 percent are inside the interval [-4, 5, 6, 9]. Thereby 564,198 segments are inapplicable, and a data set of 4,533,390 items remains.

Now the adjusted data set is analyzed for parameter loading and range estimation model parameter mix dependency. Therefore, 200 trips are selected randomly. The trip segments range between 113 and 645. So on average a trip has 450 segments (overall trip average 420 segments). For the analysis, a Leave-One-Out cross validation with $k = n = 200$ was applied. Altogether, ten runs are processed split into two test series (3 and 17 intervals). Each run-series starts with all parameter for learn and test stage. Remaining runs are done by one parameter missing. Table 1 presents the parameter loading results.

That gets the following order of importance: (1) acceleration, (2) air conditioning, (3) velocity and (4) weather. For both runs, a missing weather parameter is not significant for epsilon and maximum relative error. In contrast, the remaining parameter (1) - (3) have a strong influence on the overall result. Hence, the weather parameter could be unattended for a better calculation performance. That is crucial for the second test series which provides good results. Unfortunately, the learning process takes 65 percent longer than the first test series. So it is to deliberate about whether the increased computing time is worth it.

Table 1: Relative error in parameter loading in percentage.

parameter	all	acc.	ac	velocity	weather
3 intervals	10.3	25.4	17.4	11.9	10.4
17 intervals	6.4	25.4	15.8	12.4	6.4

Table 2: Relative error on average in percentage.

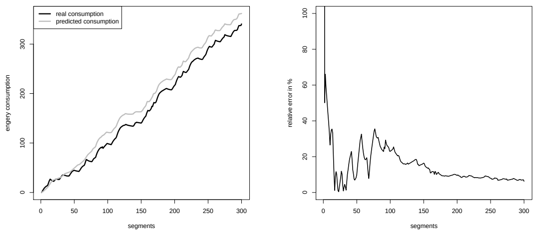
tour	segments	relative error on average			
		all param.	w/o acc.	w/o ac	w/a ac + velocity
10	4,586	9.08	29.31	19.28	27.44
20	7,970	8.42	26.52	18.83	25.32
50	19,657	7.56	26.92	18.56	25.04
100	42,087	7.87	26.86	16.40	24.47
200	81,443	7.49	26.27	15.80	23.97
500	205,909	7.66	26.92	15.22	24.01
1,000	418,344	7.52	26.61	15.26	24.07
2,000	830,729	7.63	26.95	15.35	24.17
5,000	2,086,953	7.51	26.40	15.26	23.99

Therefore, the quantity of the learning data set is crucial. To study the variation $10, 20, 50, 100, .., 5000$ trips were selected randomly for learning data sets. Thereby, it should be noted that a data set is a trip mix. In a data set of 20 items, there are 10 trips from the previous set and 10 additional trips (for 50 items 20 previous and 30 additional trips and so on). Furthermore, there is no overlapping between learning and test data sets. So starting the test runs the model has to be constructed by a learning data set. Then the relative error was detected for each test run. Table 2 presents the relative error on average.

As a result, the range estimation model converges quickly even for small data sets. For the proposed test drive the relative error with all parameters remains around 7.5 percent. Additional data sets only have a minor improvement for better quality. Without the loss of generality, the range estimation model enters the target corridor with 200 trips for any parameter combination. Furthermore, the range estimation model provides a reliable prediction of a learning data set of 200 trips.

Figure 12 presents the energy consumption forecast during a drive based on a learning data set of 200 items. For eight tours with different routes, the figure shows the off between the prediction and the real energy consumption per trip segment.

So for the whole trip the real energy tightly follows the predictive consumption (12(a)). But at tour starting prediction for single trip segments is really off



(a) (b)

Figure 12: Prediction vs real energy consumption.

(12(b)) which is not significant because at the starting point there only is minor total consumption. After a high fluctuation phase, the relative error converges straight to minor values.

8 CONCLUSIONS

SCL has implemented and installed the system as shown. The first iteration for evaluation targets two test scenarios to see those parts under realistic conditions. One of these scenarios, realized by using an acceptance test, tries to evaluate the interaction between driver and DAC. This test takes a closer look at GUI layout, the amount of shown information and usability. The results are used to optimized the DAC and to fit drivers needs as best as possible. The second scenario is focusing on data analyzes and getting feedback about how exact the range estimation model will work in case of different parameter sets and in case of small, medium and large learning data sets. A simulated scenario in combination with a huge number of drivers was used to show points of stagnation in growing data sets and the possibility to adjust the relative error by using different kind of parameters which influences the range.

The realized tests have shown some weak points. As seen in the acceptance tests participant can get in problems about appreciation in case of descriptive scenario simulations. They have to suggest the logistic scenario by adapting the instructions and answer through related questions about usability and client acceptance. Unfortunately, it's hard to use EVs and corresponding hardware installations in large scales to create test situations as realistic as possible. Apart from that, tests within the Eltrilo cabin produces data as required and can be modified to match any logistic scenario. The realistic visualization and handling within this cabin helps participants navigate those scenarios and proceed as normal. Those tests don't compensate the real test beds which has to be done within SCL as well. The simulated environment help to integrate a larger amount of participants than the test bed could yield.

Concerning the integration of knowledge base technologies, there are still open issues regarding the particular description language or logics specification. By now, we implemented the runtime assistance in the form of a Prolog rule base. The thereby implied closed world reasoning results in a good applicability of the knowledge base assistance, as well as a well-anticipated scalability with the integration of large databases. But the expressiveness of Prolog is really limited regarding the use of full description logic ontology used as top level domains. So by now, there is some kind of semantic gap between the conceptual description of higher ontologies and the until now used Prolog logic rule base. We will try to narrow this gap by testing the implementation of DL ontologies and open world based reasoning without losing the functionality, applicability and scalability.

The results about architectural drafts, analysed related projects, as well as embedded existing system landscapes in combination to the range estimation model and knowledge base for this particular domain, will be transferred into a reference architecture. This architectural documentation should help to design and implement future systems which have to be used in addition to existing logistic systems. Using a reference architecture within related projects can help to get more EVs until 2020 as intended by the German national development plan for electro mobility.

ACKNOWLEDGEMENTS

We would like to thank all members of the SCL research team here at FSU Jena – there are too many to name them all in person. We would also like to extend our gratitude to our partners within the research consortium, end users as well as research institutions and industrial developers. This project is supported by the German Federal Ministry for Economic Affairs and Energy in the IKT-II fur Elektromobilitat program under grant 01ME121(-33).

REFERENCES

Anand, N., Yang, M., van Duin, J., and Tavasszy, L. (2012). Genclon: An ontology for city logistics. *Expert Systems with Applications*, 39(15):11944 – 11960.

Apel, S., Schau, V., and Rossak, W. (2015a). Darstellung branchentypischer Abläufe und Wechselwirkungen in der innerstädtischen logistik. Technical Report Math/Inf/06/2015, *Friedrich-Schiller- Universität Jena, Institut für Informatik, Jena, Germany*.

Apel, S., Schau, V., and Rossak, W. (2015b). Geschäftsprozesse der innerstädtischen Logistik, Modellierung von allgemeinen Geschäftsprozesse. Technical Report Math/Inf/05/2015, *Friedrich-Schiller-Universität Jena, Institut für Informatik, Jena, Germany*.

Arnold, D., H.Isermann, Kuhn, A., and Tempelmeier, H. (2002). *Handbuch Logistik*. Springer, Berlin.

Beikirch, S. (2015). Mit Blaulicht und Schokolade. Internet, http://144.76.35.217/joomla/media/OTZ_11.03.2014_mit-blaulicht-und-schokolade.pdf, visited 2015-10-21.

Bundesregierung Deutschland (2012). NEP Elektromobilität. Internet, http://www.bundesregierung.de/Content/DE/Infodienst/2012/10/2012-10-12-elektromobilitaet/2012-10-12-elektromobilitaet.html, visited 2015-10-21.

Bundesregierung Deutschland (2013). IKT für Elektromobilität II. Internet, http://www.bmwi.de/BMWi/Redaktion/PDF/Publikationen/Technologie-und-Innovation/ikt-elektromobilitaet,property=pdf,bereich=bmwi2012,sprache=de,rwb=true.pdf, visited 2015-10-21.

Busch, A., Dangelmair, W., Pape, U., and Rüther, M. (2003). *Marktspiegel Supply Chain Management Systeme*. Gabler, Wiesbaden.

Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., and Stal, M. (1996). *A System of Patterns, Pattern-Oriented Software Architecture*. John Wiley and Sons.

Chappell, D. (2004). *Enterprise Service Bus*. O'Reilly Series. O'Reilly Media, Incorporated.

Chopra, S. and Meindl, P. (2007). *Supply Chain Management*, volume 3. Pearson Education, New Jersey.

Conradi, P. (2012). Reichweitenprognose für Elektromobile. In *ATZelektronik*, volume 7.

DAKO (2015). Website for the SCL project. http://smartcitylogistik.de, visited 2015-10-21.

EnBW (2015). iZEUS - Projektziele - intelligent zero emission urban system. Internet, http://www.izeus.de/projekt/beschreibung-ziele.html, visited 2015-10-21.

Faber, A. and Ammerschuber, O. (2007). Transporte im Griff. *Logistik Heute*, pages 35–37.

Gamma, E., R.Helm, Johnson, R., and Vlissides, J. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1 edition.

Gebhardt, K., Schau, V., and Rossak, W. (2015). Applying stochastic methods for range prediction in e-mobility. In *15th International Conference on Innovations for Community Services*, Nuremberg, Germany.

INNOMAN (2015). Projektidee. http://www.smobility.net/idee, visited 2015-10-21.

ISO/IEC JTC 1 (1994). *ISO 7498-1: Information technology - Open Systems Interconnection - Basic Reference Model*, 1 edition.

Jinbing, H., Youna, W., and Ying, J. (2008). Logistics decision-making support system based on ontology. In *Computational Intelligence and Design, 2008. ISCID '08. International Symposium on*, volume 1, pages 309–312.

Lian, P., Park, D.-W., and Kwon, H.-C. (2007). Design of logistics ontology for semantic representing of situation in logistics. In *Digital Media and its Application in Museum Heritages, Second Workshop on*, pages 432–437.

Logistik Heute (2005). Die Qual der Wahl. *Logistik Heute*, pages 36–37.

Navimatix (2015). Website for the eTelematik project. http://www.etelematik.de, visited 2015-10-21.

Ondruska, P. and Posner, I. (2014). The Route Not Taken: Driver-Centric Estimation of Electric Vehicle Range. In *24th International Conference on Automated Planning and Scheduling*, Portsmouth, NH, USA.

Pirron, J., Kulow, B., Hellingrath, B., and Laakmann, F. (1999). Marktübersicht SCM-Software. *Logistik Heute*, pages 69–75.

Prinz, T., Kretzschmar, J., and Schau, V. (2015). A knowledge base for electric vehicles in inner-city logistics. The Tenth International Conference on Software Engineering Advances (ICSEA).

Prümper, J. (1993). *Die Evaluation von Software auf Grundlage des Entwurfs zur internationalen Ergonomie-Norm ISO 9241 Teil 10 als Beitrag zur partizipativen Systemgestaltung - ein Fallbeispiel*. Teubner Verlag, Stuttgart.

Prümper, J. (2010). *ISONORM 9241/110-S: Evaluation of software based upon International Standard ISO 9241, Part 110*. HTW Berlin. Manuscript Questionnaire.

Schau, V., Rossak, W., Hempel, H., and Späthe, S. (2015). Smart city logistik erfurt (scl): Ict support for managing fully electric vehicles in the domain of inner city freight traffic. In *Proceedings of the 2015 International Conference on Industrial Engineering and Operations Management*, Dubai (UAE).

Technische Hochschule Deggendorf (2015). E-WALD news - Technische Hochschule Deggendorf. https://www.th-deg.de/de/forschung/projekte/e-wald/e-wald-news#nav, visited 2015-10-21.

VISPIRON CARSYNC (2015). Website for the ACM project. http://www.adaptive-city-mobility.de, visited 2015-10-21.