Journal of Traffic and Transportation Engineering 4 (2016) 177-185
doi: 10.17265/2328-2142/2016.04.001

Information Integration for Advanced Travel Information Systems

Markus C. Beutel, Sevket Gökay, Wolfgang Kluth, Karl-Heinz Krempels, Fabian Ohler, Christian Samsel, Christoph Terwelp and Maximilian Wiederhold

Information Systems, RWTH Aachen University, Aachen 52074, Germany

Abstract: ATIS (Advanced Travel Information Systems) are complex systems, dealing with a large amount of heterogeneous data from various sources. The exchange and integration of such data is therefore demanding, particularly for small mobility service providers with few IT (Information Technology) resources. To face this problem, this work illustrates an IT infrastructure to support information and service integration. On a conceptual level, this paper presents a travel information system architecture and respective information flows between components, e.g., intermodal routing, pricing and accounting. As realization, the authors developed and tested a communication adapter that enables and eases information exchange between the ATIS and heterogeneous second party service providers, e.g., carsharing operators. Furthermore, the authors developed a method of extending traditional public transportation routing using the exchanged information about sharing services. This enables travelers to query intermodal itinerary information combining public transport, bikesharing as well as carsharing services using a single application. The overall system was tested in a three-month testing phase. Initial results are promising.

Key words: Bikesharing, carsharing, e-mobility, intermodal travel, smart mobility, travel information.

1. Introduction

Combining diverse personal mobility services (e.g., traditional public transport, cabs, parking services, carsharing and ridesharing) result in better offers and better service coverage compared to individual services [1]. Eventually, this leads to a higher acceptance of alternative services (compared to using a private car) among travelers and therefore better sustainability. Because of the complexity and amount of different services, a supporting information system serving as single point of contact for travelers is required. Unfortunately, integration of such diverse travel information is a demanding task. On the one hand, mobility modes differ significantly in structure (i.e., individual transport vs. train schedule) and payment model (i.e., single ticket vs. post-paid subscriptions). On the other hand, travel information systems vary in extent and functionalities. Either way, the respective data is heterogeneous and has to be integrated effectively. Hence, this work focuses on the technological data integration of different information sources, specifically public transportation and sharing services. In order to do so, this paper refines the conceptual ATIS (advanced travel information system) architecture introduced in Ref. [2] and illustrates the workings and interaction of innovative components. Parts of this work were already presented in Ref. [3]. In addition, we further concretize the approach and describe the realization of a communication adapter which enables optimized data exchange between mobility service providers and the core ATIS. Furthermore, we introduce the x-sharing router, a specialized router module exploiting the exchanged information to extend a traditional public transport router with information about vehicle sharing systems. The overall system allows travelers to query and book of complete intermodal itineraries on a single platform. Second party mobility service providers are enabled to join the ATIS and use its resources, e.g., intermodal

Corresponding author: Markus C. Beutel, Dipl.-Kfm., research fields: business process modeling, IT-driven business models and travel information systems.

routing and accounting, with little individual integration efforts.

2. Related Work

In general, travel information provision via information systems has been investigated and improved in several areas in the last years. Crucial for the success of these information systems are two main factors: (1) the quality of the provided information in terms of comprehensiveness and completeness; (2) the systems' usability. In Ref. [4], a technology acceptance model is applied to evaluate the usability of mobile passenger information systems. According to Ref. [5], travelers are highly frustrated when using public transportation because of poor information provided, even though Ref. [6] underlines the importance of real time data provision. In Ref. [7], the authors identify and model the phases of an intermodal journey to distinguish user requirements, and in Ref. [8], a literature-based study is presented on requirements of travel information systems. The authors of Ref. [9] embedded a mobility planning model hierarchically into an agenda planning model and identified common information needs. The aforementioned works conclude that the intermodal travel information provision is lacking, which stresses the need for better travel information systems.

Modern travel information applications like Transit App[1] or Qixxit[2] differ in extent and functionalities. Above that, different research projects investigate issues related to travel information integration: for example, the IMA (intermodal mobility assistance for megacities) system is an open agent based mobility platform, that combines heterogeneous mobility services, e.g., ridesharing [10]. The European research project SUPERHUB is another open platform for urban mobility, that enables multimodal journey planning. The system incorporates intermodal routing [11] and encourages sustainable travel behavior by a behavior management component [12].

Most of these systems and applications focus rather on querying itineraries and providing journey information, but not on booking of the whole resulting journey. Either such functionality does not exist at all, or the user is assisted to switch to the mobile application of the respective provider to book tickets for each leg of the intermodal journey. From a user experience perspective, this is not desirable on multiple grounds: (1) the user makes context switches from one mobile app to others; (2) the user books multiple tickets and deals with them during her travel instead of one combined ticket for the whole journey; (3) the user has to be a customer of multiple providers to be able to book the ticket.

3. Use Cases

Using the integration method presented in Section 4, a platform is enabled to offer continuous intermodal traveling in terms of travel planning, booking, ticketing and assisting. The following use cases exemplify the advantages of information exchange and integration from a user standpoint:

- Traditional travel without data integration: Adam has a business meeting with a project partner in St. Augustin (a suburb of Bonn), Germany in the afternoon. Currently, he is planning his trip in his office in Aachen, Germany. He queries the Website of Deutsche Bahn which returns an itinerary from Aachen Main Station to Bonn Main Station. To get to Aachen Main Station he asks a colleague which bus to take and to get from Bonn to St. Augustin. He is thinking of taking a cab. Because the meeting is important and Adam knows neither the overall duration of the travel nor the price and also feels insecure about changing buses and trains, he opts for a rental car instead;
- New generation of travel supported by data integration: Bella is Adam's deputy who is in the same situation one day later. She queries the ATIS which integrates data of multiple mobility services. The system returns an itinerary involving a rental bike from the

[1] http://transitapp.com/.

[2] https://www.qixxit.de/en.

office to Aachen Main Station, trains from Aachen to Bonn and finally a reserved carsharing vehicle to St. Augustin. The system also shows the overall price and travel duration and allows to book the complete itinerary with a single click. After booking, the itinerary as well as tickets is transferred to her mobile device so she can check for train departure information easily and therefore feels secure about reaching the destination in time. Bella's travel costs less than half of Adam's and is almost as fast, while being more environment-friendly.

4. Travel Information Exchange

This section outlines the technical infrastructure proposed for data integration and the required technology.

4.1 Information Technology Architecture and System Components

Based on the general architecture presented in Ref. [2], a technical system infrastructure was created to support intermodal traveling. The current architecture is depicted in Fig. 1. The traveler interacts with the system using a mobile application or a web portal which connects to the backend using the standardized APIs (Application Programming Interfaces) URA2 and URA3. The URA (Unified Realtime API) interface family is widely used, i.e., by Transport for London.[3] The architecture of the backend resembles a distributed system consisting of different functional components and multiple internal and external endpoints.

The Dispatcher Component serves as a single endpoint for all API calls from mobile applications and the web frontend. The dispatcher re-routes and combines requests, responses and push notifications to their respective receivers. The Intermodal Router processes exhaustive door to door routing queries and serves as a merger of the different routing techniques. It is invoked by the dispatcher. The Intermodal Router is based on the RAPTOR (Round-based Public Transit Optimized Router) algorithm [13] for fast public transportation routing, using real time timetable data. The individual transport router serves pedestrian and private bicycle routing queries. It uses established routing algorithms for finding shortest paths in networks. The pricing module augments routing information supplied by the intermodal router with pricing information. The overall journey price is composed of a price per route segment which is handled individually. Prices for public transportation are modeled locally in the pricing module and can be calculated via stop numbers, distance, areas, daytime, etc. Season tickets are also considered if specified in the request. For sharing service fees, the external sharing system is inquired via the communication adapter. This allows service providers to handle pricing corresponding to their individual tariff systems. The booking module is responsible for forwarding booking requests to the respective mobility service and delivering the resulting ticket to the client. Price information is delivered to the billing module. The accounting/billing module collects all billing information from service providers to allow a real time cost check as well a monthly billing. The user management component is responsible for creating and modifying user accounts and user authorization. If required, user data is also exchanged with external services using the communication adapter.

In this work, the authors introduce the communication adapter and x-sharing router and expand on their technical details: the communication adapter is responsible for all the communication with second party mobility providers. It supports communication via all noteworthy messaging protocols: RESTful (Representational State Transfer) (using JSON (JavaScript Object Notation) or XML (Extensible Markup Language)), SOAP (Simple Object Access Protocol) and WebSocket (also using JSON or XML). The adapter exchanges travel information using a variety of open protocols i.e., IXSI (interface for x-sharing information) [14]. Besides synchronous exchange of information by request,

[3] https://api-portal.tfl.gov.uk/docs.

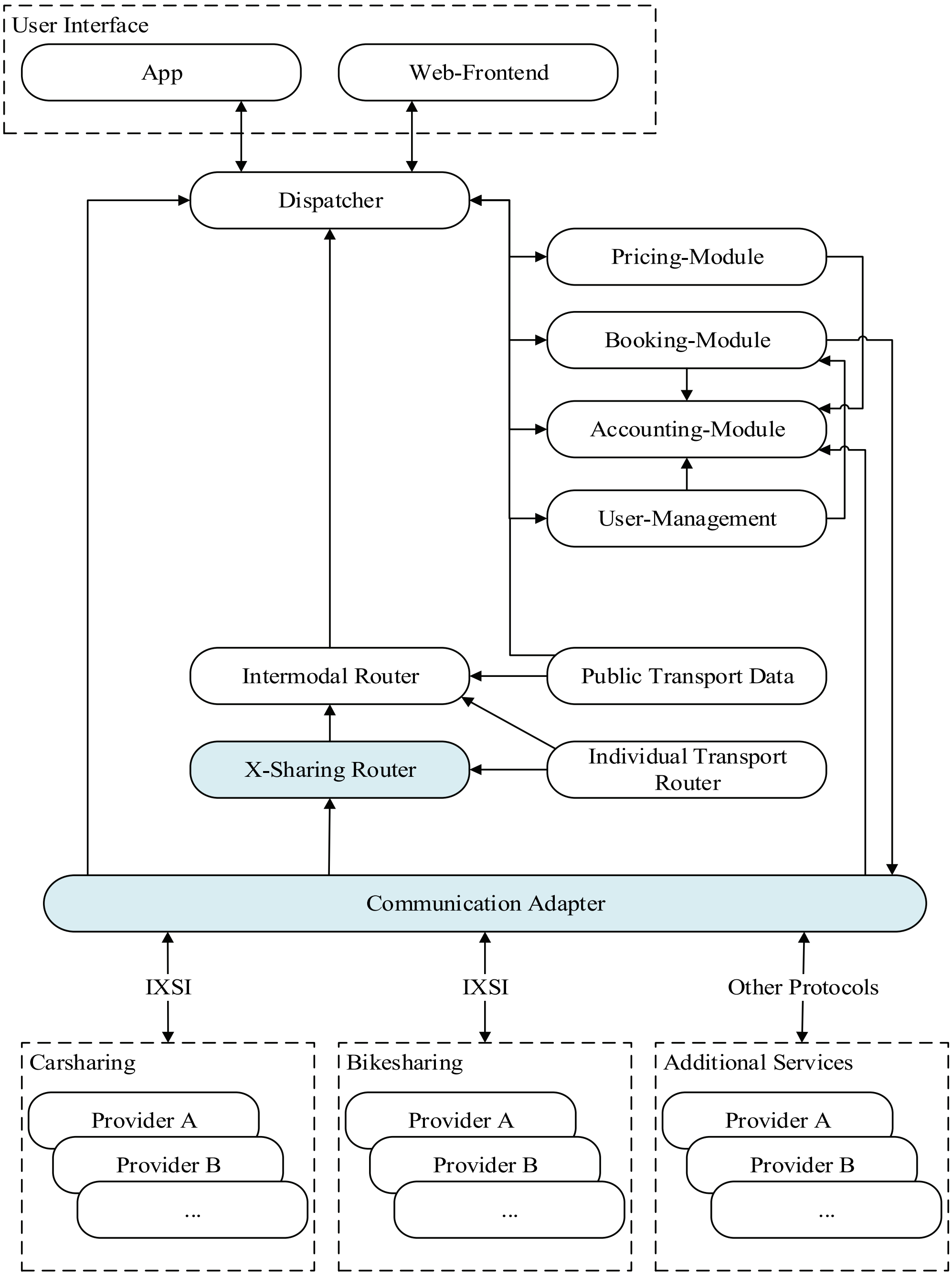

Fig. 1 System architecture [3].

IXSI allows asynchronous push notifications of vehicle sharing information, i.e., availability times.

The x-sharing router component supplements the intermodal router with routing specific to sharing services. It employs real time vehicle availability data to calculate route segments. For faster query times, legs connecting sharing places are precalculated and cached.

In order to calculate a feasible path towards the destination using individual transport, several restrictions about a realistic routing alternative must be considered. Regular carsharing, for example, usually requires the rented vehicle to be returned to exactly the same place it was taken from, which automatically prohibits this mode of transport from appearing in the middle of the journey.

4.2 Information Flow

To clarify how the different components interact with each other, this paper considers the following information flow of an intermodal travel query: Before the actual user interaction, the system has been initialized: All mobility service providers exchanged their time table information as well as vehicle availabilities and the user is registered. The user queries the system for an intermodal itinerary from Location A to Location B using a mobile app. The dispatcher forwards the request to the intermodal router which constructs multiple possible itineraries. Simplified, it does so by finding all stops and sharing places in the vicinity of both locations using the individual transport router and finding the best connections between these places using both public transport and sharing service (computed by the x-sharing router). In the next step, these itineraries are forwarded to the pricing module which calculates and annotates prices for every route segment. The list of itineraries, including prices, is then supplied to the app, namely to the user. If the user opts to book a specific itinerary, he or she has to authenticate him- or herself which is handled by the user management. Using the authentication token (supplied by user management) and unique identifier for desired itinerary (supplied by the routing), the booking commences: the booking module informs mobility services via the communication adapter component to book the service and forwards the resulting billing information to the accounting module and the booking confirmation/ticket to the client. Now all preparation steps for the travel are complete and the user can conduct the travel. During the travel, he or she is notified about changes/updates in her subsequent route segments. Directly after the travel or alternatively at the end of the accounting period, the user is billed by the accounting module with the assistance of the user management (for billing information).

5. Realization

This section presents our implementation of newly-introduced components listed in the previous section.

5.1 Communication Adapter

The communication adapter, from a technical standpoint, is a Java EE 7 application, consisting of many modules and running on a WildFly 9 [4] application server. The IXSI module is a WebSocket client and leverages the Java API for WebSocket (JSR 356). WebSocket is a TCP (Transmission Control Protocol) based protocol that enables bi-directional communication between two parties omitting the conventional client-server characteristic. Therefore, it serves as a basis to build message exchange patterns like request-response and publish-subscribe, both of which are necessary for IXSI. Since the WebSocket connections are long-lived, there is also less transport overhead. The technical realization of the communication adapter component is depicted in Fig. 2. The working principle of the adapter is described as an example using IXSI (Interface for X-Sharing Information) [14]. The complex structure of this protocol makes it challenging to employ adequately and requires a design that covers a lot of ground. Because of this, the authors presume the adapter realization to be general enough to be applied to many, if not all, mobility service protocols. Since WebSocket is a low-level transport protocol and IXSI does not provide a framework to handle messages, the authors had to come up with an architecture that has a clear processing pipeline in which every stage only performs a dedicated part of the job and forwards the job to the next stage. That allows to follow the separation of concerns design principle.

Another goal is to abstract the underlying messaging concerns from the business logic. In order to achieve this, the authors use ideas and notions from message brokers and organize their architecture as follows: a

[4] http://wildfly.org/.

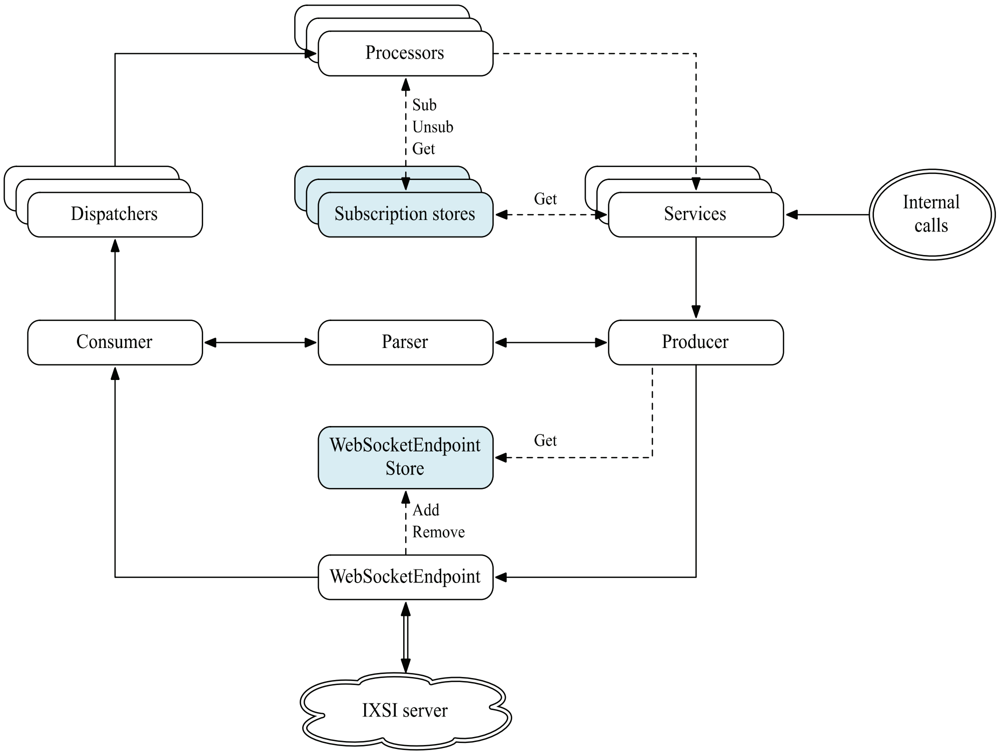

Fig. 2 Communication adapter.

WebSocket endpoint contains all WebSocket-related logic (e.g., regarding the life cycle of the communication, receiving and sending messages). The parser's task is to serialize Java objects into the textual representation for outgoing messages, and to deserialize incoming messages into Java objects. It also validates every incoming/outgoing message against the respective XML schema to ensure correctness. For incoming messages, the consumer is merely the gateway between the endpoint and dispatchers which utilizes the parser to deserialize the payload. Dispatchers handle messages which are hierarchical with one root element containing one of the multiple types of messages. This requires a decision tree with a dispatcher at every level of hierarchy that delegates to the next one. At the lowest level, the message is passed to one of the processors depending on the type of the message. A processor contains the actual business logic to process the message (e.g., database operation, internal or external component call). Depending on the message type, it may be necessary to communicate with the server as a result of an operation. The processors call one of services to realize that. The services provide an abstraction for building and sending messages and can be used by processors or any other component. The hierarchical nature [13] of IXSI suggests using a service for each level. For outgoing messages, the producer determines which partner system to send the message to (and thereby which endpoint), serializes the Java object utilizing the parser and passes it through to the endpoint for the actual transport-level transmission of the payload.

Besides, this paper introduced another concept to store information in memory that is subject to frequent change and not worth persisting in our relational database: WebSocket endpoint store keeps track of open connections to partner systems. By this means, this article aimed to support: (1) multiple connections to a single server in order to prevent unexpected concurrency issues and distribute load; (2) multiple servers and therefore different mobility providers. Subscription stores contain references to data objects which can be subscribed to in order to get push notifications from the server about their status changes.

In this way, the server always keeps the client up-to-date with minimal delay.

Since IXSI and most other mobility service protocols are designed to support many use cases but the providers have varying capabilities or want to integrate only a part of their systems due to business decisions, the adapter is capable of enabling/disabling protocol features (i.e., services) for a provider based on the contract between the both parties. The lack of a handshake mechanism in the IXSI specification to agrees on what services to use led to fulfill the requirement on the implementation level. This is realized by storing the complete configuration for each mobility provider in database, reading it in initialization phase and only allowing interactions that are part of activated features during run time.

5.2 X-Sharing Router

Our x-sharing router component extends an existing implementation of the RAPTOR algorithm [13]. RAPTOR relies on a system of—as the name suggests—round-based traversal of public transportation routes and, in contrast to previous attempts, abandons the practice of representing the network in a graph structure. Instead, RAPTOR works with adjacency lists and pointers as the data structure of choice and operates on a paradigm stemming from dynamic programming. In addition to optimizing routes for the earliest arrival problem, the algorithm aims for minimizing the number of transfers users have to undertake on a journey. By iteratively traversing routes and deciding at each round whether to switch routes, the algorithm decides on an optimal sequence of trips for reaching the destination at the earliest possible time. In order to minimize computation effort, the algorithm employs a label, the earliest known arrival time for each stop. Routes are then traversed more efficiently by asserting their reachability based on these labels, yielding a performance that is up to orders of magnitude faster than traditional algorithms. Consequently, it is employed in various public transport routing products.

In order to support the additional modes of transport which were described in the previous sections, the restrictions of RAPTOR are circumvented with a preprocessing-based route cache to extend the transit network used, while retaining its high performance. The technical base of the x-sharing router is similar to the communication adapter component, being a Java EE 7 application running on WildFly 9. All possible routes between sharing locations (i.e., carsharing stations) and potential destinations in the coverage area are precalculated based on a configurable coordinate grid. For calculation, a standard, shortest path, Dijkstra-based algorithm is used.

The x-sharing router accepts different types of requests: minimal, compact and detailed. Minimal requests are used while the routing algorithm is running, whereas compact and detailed requests are follow-ups in case the itinerary is a candidate and to be presented to the traveler. The respective responses to those requests contain additional information, e.g., regarding the vehicle and sharing station. Minimal requests on the other hand only contain the information relevant to the routing algorithm: coordinates for start and destination, time of departure (or of arrival if desired), and mode of transportation. The response contains the travel duration, number of changes and a trip ID for later reference (i.e., a follow-up detailed request). By applying a Euclidian distance function to both supplied coordinates, the system finds the nearest grid points for which a priority list of nearest sharing stations is kept. While traversing the priority list of stations, the router checks for the presence of available vehicles at stations (during the runtime of the router, it routinely synchronizes with the communication adapter component database to update the availability of vehicles at each station). When a pair of appropriate stations is found, the precalculated route is fetched from the database and the result is returned to the RAPTOR router. For efficiency, all requests can be batched and will then be processed in parallel.

Fig. 3 Field test [15, 16].

Performance tests showed that that single minimal requests have response times in the magnitude of 1 ms, which leads to overall query times of the intermodal routing of 10~30 s.

6. Discussion and Conclusions

The authors contributed a conceptual architecture and depicted the functional information exchange between system components. Above that, the authors also presented the prototypical implementation of the communication adapter component, which enables mobility service providers to easily connect to the core system, and the x-sharing routing, which allows fully intermodal travel information. The solution allows participation of second party mobility service providers based on standardized interface specifications.

Users benefit from a centralized information system that allows convenient and comprehensive intermodal travel information provision. The system was evaluated from users' perspective in a comprehensive field test in Refs. [15, 16] (Fig. 3) and has been considered as useful by the participants. From providers' perspective, the main advantage of this solution is the technological accessibility of the core system towards second party providers with limited integration efforts. The paper showed that the approach is technically feasible by successfully integrating three different second party sharing services, one bike sharing system and two carsharing systems, with a public transport operator on a single platform. These providers benefit from centralized user interfaces, intermodal routing and accounting functionalities without implementing them individually.

Some drawbacks might occur concerning the data management. Sensitive data, e.g., user information is located and aggregated at the centralized core system—instead of separate provider databases. In addition, consolidating distribution channels of various providers might result in high market transparency and, as a consequence, in rising competition. Although standardized interfaces and architectures simplify data exchange, the implementation may be demanding for mobility service providers, especially if they are still using old-fashioned IT systems. Also, it is unknown how well the system performs with higher number of participating mobility services.

References

[1] Papangelis, K., Sripada, S., Huwer, U. 2004. "Public Transport and Carsharing—Benefits and Effects of Combined Services." *Transport Policy* 11: 77-87.

[2] Beutel, M. C., Gokay, S., Kluth, W., Krempels, K.-H., Samsel, C., and Terwelp, C. 2014. "Product Oriented Integration of Heterogeneous Mobility Services." In *Proceedings of the 17th International Conference on Intelligent Transportation Systems (ITSC),* 1529-34.

[3] Beutel, M. C., Gökay, S., Kluth, K., Krempels, K.-H., Samsel, C., and Terwelp, C. 2015. "Wiederhold. Heterogeneous Travel Information Exchange." Prensented at 2nd EAI International Conference on Mobility in IoT (MobilityIOT 2015), Rome.

[4] Beul-Leusmann, S., Samsel, C., Wiederhold, M., Krempels, K.-H., Jakobs, E.-M., and Ziefle, M. 2014. "Usability Evaluation of Mobile Passenger Information Systems. Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience." *Lecture Notes in Computer Science* 8517: 217-28.

[5] Corsar, D., Velaga, N., Edwards, P., and Nelson, J. D.

2013. "Developing a Real Time Passenger Information System for Rural Areas." In *Human Interface and the Management of Information. Information and Interaction for Health, Safety, Mobility and Complex Environments,* edited by Yamamoto, S. Berlin: Springer Berlin Heidelberg.

[6] Garcia, C. R., Candela, S., Ginory, J., Quesada-Arenciba, A., and Alayon, F. 2012. "On Route Travel Assistance for Public Transport Based on Android Technology." In *Proceedings of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing,* 840-5.

[7] Digmayer, C., Vogelsang, S., and Jakobs, E. 2015. "Designing Mobility Apps to Support Intermodal Travel Chains." In *Proceedings of the 33rd Annual International Conference on the Design of Communication,* 1-11.

[8] Vogelsang, S., Digmayer, C., and Jakobs, E. 2015. "User Requirements on Intermodal Traveler Information Systems." In *Proceedings of the 2015 IEEE IPCC (International Professional Communication Conference),* 1-9.

[9] Wienken, T., Mayas, C., Hörold, S., and Krömker, H. 2014. "Model of Mobility Oriented Agenda Planning. " In *Proceedings of the 16th International Conference on Human-Computer Interaction. Applications and Services,* 537-44.

[10] Keiser, J., Masuch, N., Lützenberger, M., Grunewald, D., Kern, M., and Trollmann, F., et al. 2015. "IMA—An Adaptable and Dynamic Service Platform for Intermodal Mobility Assistance." In *Proceedings of the 17th IEEE International Conference on ITSC,* 1521-7.

[11] Hrncir, J., and Jakob, M. 2013. "Generalised Time-dependent Graphs for Fully Multimodal Journey Planning." In *Proceedings of the 16th International IEEE Conference on ITSC,* 2138-45.

[12] Wells, S., Forbes, P., Masthoff, J., Gabrielli, S., and Jylha, A. 2013. "SUPERHUB: Integrating Digital Behaviour Management into a Novel Sustainable Urban Mobility System." In *Proceedings of the 27th International BCS on Human Computer Interaction Conference.*

[13] Delling, D., Pajor, T., and Werneck, R. F. 2014. "Round-Based Public Transit Routing." *Transportation Science*: 1-14. http://dx.doi.org/10.1287/trsc.2014.0534.

[14] Kluth, W., Beutel, M. C., Gökay, S. , Krempels, K. H., Samsel, C., and Terwelp, C. 2015. "IXSI-Interface for X-Sharing Information." In *Proceedings of the 11th International Conference on Web Information Systems and Technologies,* 293-8.

[15] Beutel, M. C., Zaunbrecher, B. S., Himmel, S., Krempels, K. H., and Ziefle, M. 2016. "Evaluation of an Integrated Intermodal Travel Service." In *Proceedings of the 5th International Conference on Smart Cities and Green ICT Systems,* 363-71.

[16] Himmel, S., Zaunbrecher, B. S., Ziefle, M., and Beutel, M. C. 2016. "Chances for Urban Electromobility Field Test of an Intermodal Travel System and Effect on Usage Intention." In *Proceedings of the 18th International Conference on Human-Computer Interaction,* 472-84.