IXSI - Interface for X-Sharing Information

Wolfgang Kluth, Markus C. Beutel, Sevket Gökay,
Karl-Heinz Krempels, Christian Samsel and Christoph Terwelp
Information Systems, RWTH Aachen University, Aachen, Germany

Keywords: Travel Information, Intermodal Transportation, System Integration, Carsharing.

Abstract: The increasing demand for mobility, especially for individual transport, leads to more pollution, congested cities and shortage of parking. New ways of mobility could mitigate these issues. Unfortunately, such forms of mobility, i.e. carsharing, are usually isolated services, because of the missing integration with other mobility modes. Our aim is to offer a combination of heterogeneous services on a single platform. We argue that such an integration allows optimal offers and higher usability and therefore results with a higher acceptance among travelers. Embedding a vehicle rental system into a travel information system information-wise is a step forward. For this purpose, we developed an interface for x-sharing information, short IXSI, specialized in connecting vehicle rental systems with a travel information system. IXSI is an XML-based, B2B interface with functions for, e.g., exchanging basic vehicle data and price information. The interface enables travel information systems to perform bookings of carsharing and bikesharing vehicles and therefore allows the customer to use traditional public transport services as well as rental services seamlessly. Furthermore, we briefly present our IXSI implementations on travel information and vehicle rental system side.

1 INTRODUCTION

Carsharing is a growing market worldwide (Hayashi et al., 2014). In the past decade, the number of shared vehicles, as well as the number of members using carsharing have increased rapidly (Shaheen, 2012). Forecasts predict an extensive growth of carsharing users from 700,000 in 2011 up to 15 million in 2020 in Europe (Statista GmbH, 2015). Nevertheless, carsharing is usually seen as a supplement to traditional public transportation services like trains or underground railways in order to improve service coverage (Huwer, 2004). That is to say, a customer could travel to a different city by train and switch to a carsharing vehicle to reach his or her final destination which might not be reachable via public transport.

Although such a journey is already possible today by manually booking a train ticket as well as a carsharing service, a single point of contact for planning, booking and assisting a trip would greatly simplify the journey. To render such a single point of contact possible, system integration is required. Unfortunately, due to the heterogeneity of different mobility modes, e.g., train schedule vs. individual transport, system integration efforts are demanding.

For the following remarks, we define a travel information system (TIS) as a system covering (travel) data integration, routing, price calculation, booking, and the presentation of information through a user interface. This system is supposed to be the SPOC for all travel related matters. Assisting the customer while traveling is also conceivable. Moreover, we define a vehicle rental system (VRS) as an information system responsible for administration and accounting of rental vehicles and customers. Rental vehicles in this context can be cars (with conventional engine or electric) or bikes (with or without electric drive).

Hence, we contribute an approach, interfacing TIS and VRS under consideration of different procedures, functionalities and requirements.

This work is structured as follows. First, Section 2 summarizes related work in the field of interest. As a focal point, Section 3 describes the contributed approach and illustrates functionalities by means of relevant use cases. Afterwards, Section 4 covers the actual information exchange in detail and Section 5 presents our implementation efforts. Finally, Section 6 reflects the approach and describes future work.

2 RELATED WORK

In the particular area of travel information systems,

Kluth W., C. Beutel M., Gökay S., Krempels K., Samsel C. and Terwelp C..
IXSI - Interface for X-Sharing Information.
DOI: 10.5220/0005506102930298
In *Proceedings of the 11th International Conference on Web Information Systems and Technologies* (WEBIST-2015), pages 293-298
ISBN: 978-989-758-106-9

system and data integration is a prominent research field (Figueiredo et al., 2001).

(Fluhr and Lutz, 2011) define special use case types for electric vehicles and underline their importance for the development of comprehensive information system architectures. Similarly, our work starts out from outlining the use cases.

(Barth et al., 2002) focus on wireless communication of shared vehicle systems and describe a data communication architecture for sharing systems with multiple stations. (Beutel et al., 2014) propose an architecture for comprehensive travel information systems and touch on the issue of interfacing heterogeneous mobility modes. Thereby, methods of data integration are addressed.

Furthermore, (Doshi, S., 2010) focuses on the development of a multimodal traveler information system and describes an interface with a limited amount of functionalities.

3 USE CASES AND APPROACH

Integration of different mobility providers into one information system which provides a central interface to the user is challenging. Current methods which involve redirecting users from the TIS to the VRS for further information and booking services, lowers the usability and quality of the process.

Therefore, a protocol is needed which allows a seamless integration of one or more VRS into a single TIS which handles vehicle information, bookings, and price information.

3.1 Use Case Evolution

To investigate possible effects, two use-cases are presented, with and without VRS integration, taking into consideration the roles represented in Figure 1.

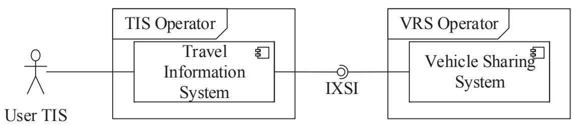

Figure 1: Overview Roles.

Without Integration

Today, Jack has a business appointment with a new customer to discuss a project in Cologne later that day. At the moment, he is sitting in his office in Aachen, so he needs to plan a trip from Aachen to Cologne. For this purpose, he uses a TIS to request several travel chains. The results represent solutions, using public transport, but no VRS options. To get further options, e.g. car sharing, he has to explicitly query a VRS to get information about the availability of nearby stations and cars. Jack has decided to take the car sharing option, despite the fact that he has to walk one kilometer to the next car station. He stops further efforts to find a better solution, because combining public transport and car sharing is too complex and time-consuming.

With Integration

Jack's colleague, Linda is in the same situation, but her appointment is one day later. She has access to a novel TIS with a full integration of the VRS. As input for her request she is using her office address as the starting point, the customer's address as the destination, the arrival, and return time. As result she gets a list of trips with public transportation and car sharing, separated and combined. She decides to take the route which involves traveling by bus to the carsharing station for the first kilometer. From there she is going to drive the carsharing vehicle the rest of the way. She is not worried about her plans, because the TIS checked implicitly the availability of the car and in the next step she can book it directly without involving the VRS.

Interlinking Between TIS & VRS

Considering both use cases, the integration of the VRS into the TIS shows significant improvements of usability. One fundamental piece is the integration of the VRS into the TIS, which covers all necessary functionalities (e.g., availability query and vehicle booking). For this purpose, IXSI (Interface for X-Sharing Information), which connects the VRS with the TIS, was developed, In the following section the functionality and general structure of IXSI are presented.

3.2 Hierarchy Model

The hierarchy model of IXSI describes different qualities of information coupling based on service groups. Furthermore, it contains recommendation for various levels of implementation.

The interlinking between TIS and VRS needs at least the implementation of Service 1 (static data) by both systems. Figure 2 shows the dependencies between services.

Service 1 (static data) exchanges information about providers and static data of booking targets (see Subsection 4.1). For example, sharing stations (VRS) can be presented in a TIS.

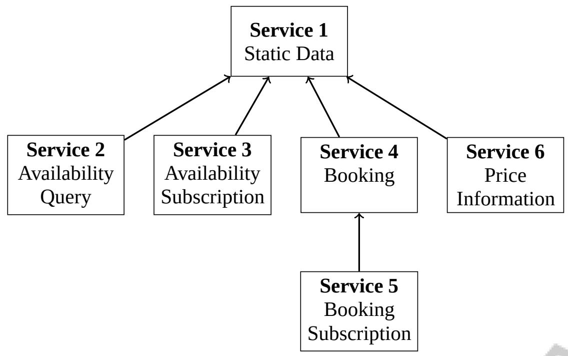

Figure 2: IXSI Service Groups dependencies.

Service 2 (availability query) provides queries for availability information. When a customer is interested in an itinerary containing a specific vehicle, the actual availability times of the respective booking target are requested to ensure the vehicle is available.

Service 3 (availability subscription) depends on Service 1 and is responsible for an asynchronous exchange of availability information. The TIS can subscribe to booking targets in order to avoid multiple requests. After subscribing to a set of booking targets, the VRS continuously informs the TIS about the availability status. This service is an addition to Service 2 which allows the information exchange to scale up by preventing necessary messages and also potentially increases the interactivity of the service by preventing delays.

Service 4 (booking) depends on Service 1 and allows vehicle booking, and changing or cancellation of those by the TIS on behalf of the (VRS) customer.

Service 5 (booking subscription) depends on Service 4 and provides a subscription to booking updates. The TIS can subscribe to existing bookings in order to receive updates from the VRS if the booking has changed (i.e., vehicle is not available anymore).

Service 6 (price information) depends on Service 1 and provides price information of rental services. Based on the starting point, destination, start time, and end time, the TIS can request price information (individualized for a customer or not) from the VRS, which responds with a tentative price and if necessary with individual items.

4 INFORMATION EXCHANGE

This section gives a brief overview about how the information exchange between travel information system and vehicle sharing system is actually conducted.

4.1 Data Model

To understand the information flow, understanding of how real world properties are modeled is required. This work does not contain a complete list of all types and attributes, but instead outlines the relation between the exchanged objects among themselves and the real world. Please refer to the specification[1] for more details.

Place describes a location where rental vehicles are available for rent or to return. Typical examples are bike sharing or car sharing stations.

Booking Target is an abstract representation of one or more vehicles which can be rented. It usually denotes a group of similar characteristics. The reason for the abstraction is that the provider can replace a vehicle with another at any time.

Availability is a time span during which a booking target is free for booking.

Booking is a binding reservation of a booking target for a specific time span.

4.2 Information Flow

The following UML-like sequence diagrams give an overview of a possible information exchange using IXSI. In the depicted information exchange, the user inquires the TIS for a travel information, books a specific itinerary and subscribes to notifications. As shown in the different diagrams, static information is exchanged as initialization (i.e., places), asynchronously to inform about updates (i.e., booking targets), and synchronously (i.e., bookings). Although complex, these different ways of exchanging data, depending on the purpose, allow efficient as well as low-delay communication.
As initialization the VRS conveys the booking targets to the TIS and TIS subscribes to changes of those (see Figure 3). This happens proactively without any user involvement. As soon as a booking target changes, the VRS informs the TIS asynchronously.
For the actual travel information (Figure 4), the user makes a door to door request, using, e.g., a smartphone. For the respective itineraries, multiple

[1]https://github.com/RWTH-i5-IDSG/ixsi

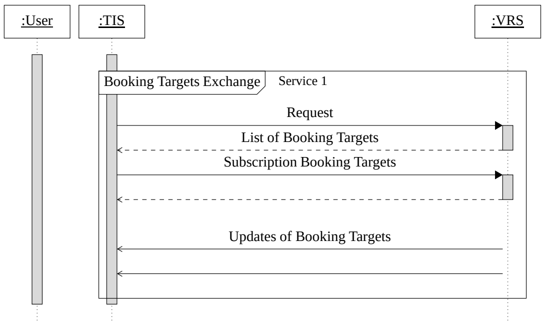

Figure 3: Interaction Sequence: Booking Target Exchange.

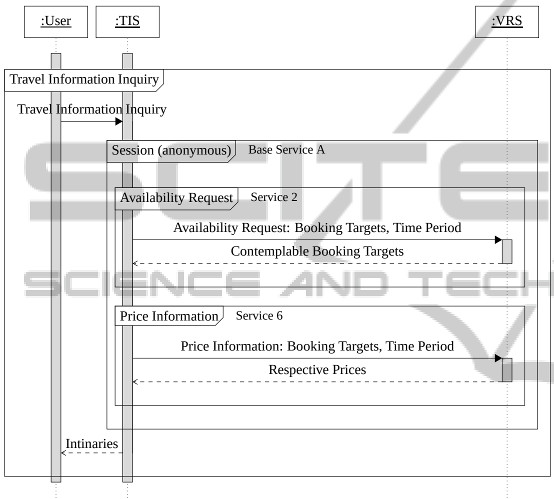

Figure 4: Interaction Sequence: Travel Information.

rental vehicles are possible, for which the TIS synchronously checks the availability. Furthermore, the prices are obtained. All in all, the TIS returns the possible itineraries and overall prices to the user device. In the next step (Figure 5), the user selects a suitable itinerary and wants to book it. He logs into the mobile app using his personal credentials. These credentials are passed by the TIS to the VRS to create a session (usage of an access token is omitted for simplicity). During the session, the actual booking for the vehicle is conducted. To book a vehicle, the TIS requests a booking using a booking target reference and a time span proposal, which is answered with a booking confirmation and the actual booked time. Finally, the TIS informs the user about the successful booking.
Optionally, the TIS can subscribe to booking changes (e.g., cancellations). This is used to inform the user about such events and to automatically offer alternatives (Figure 6).

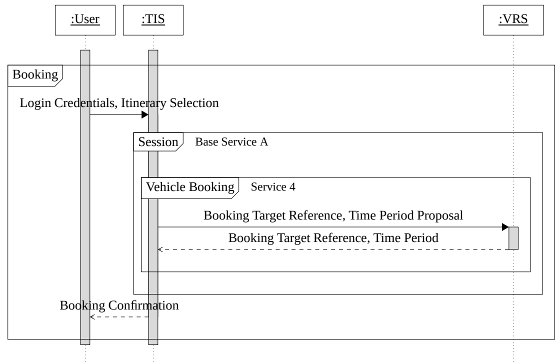

Figure 5: Interaction Sequence: Booking.

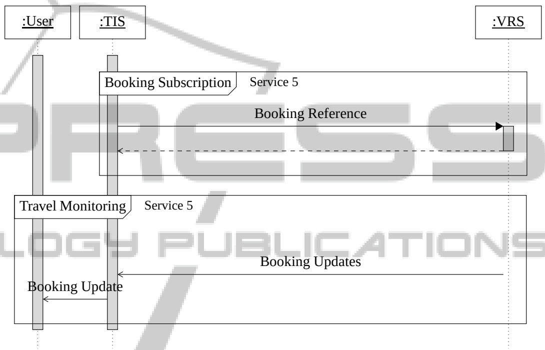

Figure 6: Interaction Sequence: Travel Monitoring.

4.3 Message Format, Message Coding and Transport

Messages between TIS and VRS are transferred as XML according to a standardized XML Schema. Messages belong to one of five archtypes, *Request*, *Response*, *SubscriptionRequest*, *SubscriptionResponse* and *SubscriptionMessage*. *Request* and *Response* form the identically named information exchange pattern, whereas the latter three allow a asynchronous data exchange as depicted in, i.e., Figure 3.

As the interface allows asynchronous subscription besides the usual request/response scheme, usage of the WebSocket protocol instead of plain HTTP is recommended. WebSockets allow persistent connections between both systems and a bidirectional message exchange. See (Pimentel and Nickerson, 2012) for a comparison of HTTP and WebSockets.

4.4 Security and Encryption

IXSI is a B2B interface and therefore does not include an internal authentication mechanism. Instead the communication partners are advised to use existing authentication mechanisms such as SSL certifi-

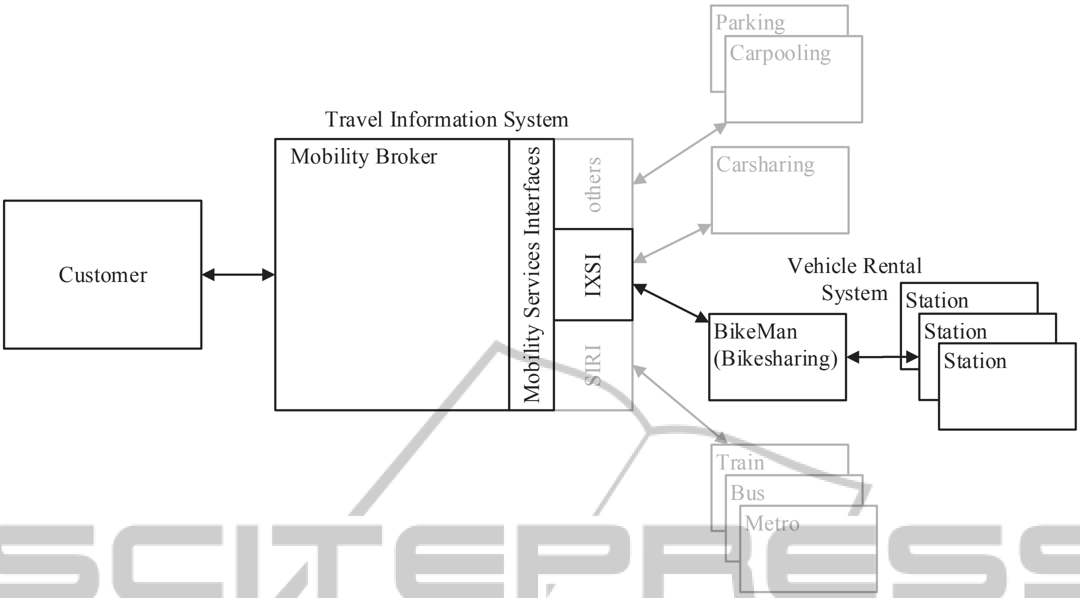

Figure 7: Overview Mobility Broker Architecture.

cates or a virtual private network (VPN) authentication.

To ensure confidentiality of the transferred data, encryption is required. Both the SSL/TLS security layer and VPN are suitable.

5 IMPLEMENTATION

In the previous sections we presented the purpose, structure and processes of IXSI. In addition to the theoretical work, we implemented IXSI for our research project *Mobility Broker* (Beutel et al., 2014). Mobility Broker is an information system which integrates multiple heterogeneous mobility services including - but not limited to - buses, bikesharing and carsharing, into one joint platform. As shown in Figure 7 Mobility Broker communicates with service providers using different proprietary and standardized B2B interfaces (IXSI being one of them) and with customers (respectively their applications) using open APIs for web and mobile application development.

From a technical standpoint, it is a Java EE 7 application consisting of many modules and running on a WildFly[2] application server. The IXSI module is a WebSocket client and leverages the Java API for WebSocket (JSR 356) implemented by the Undertow[3] web server, which is part of WildFly.

As for the VRS, we chose to extend our open source tool *BikeMan*[4] with the IXSI implementation. BikeMan is a station-flexible e-bike/pedelec sharing system which enables the management of stations, pedelecs and customers. Moreover, it allows to monitor pedelec rentals. BikeMan is a Java web application based on Spring Boot[5] and published under Apache License, Version 2. In the interaction with Mobility Broker, BikeMan acts as a WebSocket server and uses Spring's own WebSocket API which is JSR 356 compatible.

Since WebSocket is a low-level transport protocol, it does not provide the following messaging conveniences out of the box:

1. RPC-style synchronous messaging: WebSocket is asynchronous and only covers exchanging of data or text frames, but not communication with request-response pattern.
2. Message routing: It does not interfere with the interpretation of the content, namely the determination of the message type and which procedure to subsequently invoke.
3. Error handling: It does not offer any mechanism for reliable message delivery or fault-tolerant connectivity.

Therefore, we had to tackle some challenges on both sides and build our own abstractions to deal with the high-level interface that is IXSI.

[2]http://wildfly.org/
[3]http://undertow.io/
[4]https://github.com/RWTH-i5-IDSG/BikeMan
[5]http://projects.spring.io/spring-boot/

6 DISCUSSION AND OUTLOOK

The aim of this paper was to present our approach of integrating sharing/rental vehicles into an intermodal journey by connecting the related IT systems. To do so, an asynchronous, XML-based interface was created. Our presented implementation shows that technology-wise an integration is complex, but possible. Unfortunately, the technical implementation is only the first step of offering an integrated mobility service to citizens. There are still political, economical and sociological challenges involved and might even contain show-stoppers, which could potentially render the technical solutions pointless. Furthermore, the interface as well as the implementation are not in productive use and require more investigation. Specifically, performance and stability will be focal points.

We are currently in the phase of deploying our prototype to a closed user group to identify further potential problems in technical and other aspects. After the system operation has been assured, user studies will commence to quantify the acceptance of a system, integrating sharing/rental systems and traditional public transports.

As a final step of development, we plan to standardize IXSI in close collaboration with a suitable international organization.

ACKNOWLEDGEMENTS

IXSI was developed in cooperation by Cantamen GmbH, HaCon Ing.-Ges. mbH and RWTH Aachen University for project eConnect Germany. The authors would like to thank Peter von Grumbkow, Dirk Hillbrecht, Heike Twele and Gerhard Wagner for their work on IXSI. This work was funded by German Federal Ministry of Economics and Technology for projects eConnect Germany (01ME12052) and Mobility Broker (01ME12135A).

REFERENCES

Barth, M., Xue, L., Chen, Y., and Todd, M. (2002). A hybrid communication architecture for intelligent shared vehicle systems. *IEEE Intelligent Vehicle Symposium (Vol. 2)*, pages 557–563.

Beutel, M. C., Goekay, S., Kluth, W., Krempels, K.-H., Terwelp, C., and Samsel, C. (2014). Product Oriented Integration of Heterogeneous Mobility Services. *IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1529 – 1534.

Doshi, S. (2010). Designing a Multimodal Traveler Information Platform for Urban Transportation.

Figueiredo, L., Jesus, I., Tenreiro Machado, J., Rui Ferreira, J., and Martins de Carvalho, J. (2001). Towards the Development of Intelligent Transportation Systems. *IEEE International Conference on Intelligent Transportation Systems (ITSC)*.

Fluhr, J. and Lutz, T. (2011). Use Case Types for Communication with and for Electric Vehicles (EV). *2011 17th International Conference on Concurrent Enterprising (ICE)*, pages 1–6.

Hayashi, T., Hidaka, K., and Teshima, S. (2014). Carsharing and IT Enabled Services. *2014 Annual SRII Global Conference*, (Fig 1):274–280.

Huwer, U. (2004). Public transport and csar-sharingbenefits and effects of combined services. *Transport Policy Vol. 11*, pages 77–87.

Pimentel, V. and Nickerson, B. (2012). Communicating and displaying real-time data with websocket. *Internet Computing, IEEE*, 16(4):45–53.

Shaheen, S. (2012). Understanding from Shared-Use Mobility Research.

Statista GmbH (2015). Anzahl der Carsharing-Nutzer in Europa in den Jahren 2011 und 2020 (in Millionen).