An Ontology for Service Semantic Interoperability in the Smartphone-Based Tourist Trip Planning System

Sergei Mikhailov[1,2], Alexey Kashevnik[2,3]
[1]ITMO University, Saint Petersburg, Russian Federation
[2]SPIIRAS, Saint Petersburg, Russian Federation
[3]Petrozavodsk State University (PetrSu), Russian Federation
sergei.mikhailov@iias.spb.su, alexey@iias.spb.su

***Abstract*—This paper presents an ontology-based approach for semantic interoperability tourist trip planning services. The proposed ontology describes a tourist, an attraction route and context information about tourist and his/her environment. This ontology is developed within the Tourist Trip Planning System, which consists of a set of interacting services. All services work accordingly to the proposed ontology which leads to service semantic interoperability and allows to increase interaction speed between them.**

I. Introduction

The tourism is one of the profitable spheres of the world economy. According to the UNWTO World Tourism Organization, the total number of international tourist trips and the international tourism income are constantly increasing. The electronic technologies usage in tourism leads to the emergence of new challenges and opportunities in this area. There is a substantial development of research on IT and tourism spanning over twenty years [1]. One example of the interaction of electronic technology and tourism is the massive use of e-tourism services [2]. This type of services provides tourist support in unfamiliar environments by providing various services, such as a trip planning based on tourist preferences, showing information about nearby POIs (points of interest), etc. A relatively recent development in e-tourism lies in the use of mobile devices as a primary platform for information access, giving rise to the field of mobile tourism [3].

This paper presents the Tourist Trip Planning System which is based on the previously developed Tourist Assistant — TAIS [4]. This system is designed as a service-oriented architecture which provide attraction visiting route and information about attraction to a tourist. The authors offers to use a context-oriented approach for constructing tourist attraction visiting route. An ontology based approach is used for the data and knowledge organization [5]. This approach allows one to overcome the knowledge semantic heterogeneity obtained from various sources. The proposed ontology describes a tourist and his/her preferences, a city in which the tourist is located, a constructed tourist trip route between attractions and context situation, which can be divided into tourist part and environment part (city and attraction context information). The ontology has been constructed based on concepts of ontologies FOAF (http://www.foaf-project.org/) and GEONAMES (http://www.geonames.org/ontology/documentation.html). Each service works accordingly to the proposed ontology thereby skipping an ontology matching phase in a smart space, which enables a semantic interoperability between services. The ontology is represented by RDF triples since the system utilises the Smart-M3 information sharing platform [6] operated with RDF.

The following paper is structured as follows. Section II describes the current state of attraction recommendation system area. Section III give description of proposed tourist trip planning system concept by showing a use case scenario (III-A) and related models — the system reference model (III-B) and the ontology model (III-C). Section IV describes the ontology usage within the tourist trip planning services and their interaction. Finally, section V gives a conclusion.

II. Related work

This section describes the current state of tourist trip planning and recommendation systems and algorithms used in this kind of systems.

In the paper [8] authors present the "Scenic Athens" — a context-aware mobile city guide for Athens (Greece). In this solution tourist trip planning algorithm works with POI's (points of interest) and scenic (walking) routes among a city. The authors claim that tourist usually prefers to walk through pedestrian zones, cultural or urban zones rather only getting shortest and fastest way of reaching POIs. The iterated local search algorithm is using for solving mixed team orienteering problem with time windows. For route constructing the weather and tourist pace and preferences can optionally be taken into account.

The paper [7] describes personalized location-based mobile tourism application. This application is based on the iBike system in Taichung City, Taiwan. A hybrid filtering technique is used to collect tourist information. The adaptation of the ant colony optimization algorithm was used as route constructing mechanism taken for more precise adjustment of geolocation recommendations to tourists. The authors use the technology acceptance model and the information systems success model for users intentions investigating to further using in the developed system. The technology acceptance model comprises three dimensions: perceived ease of use, perceived usefulness, and usage behavioural intention. The information systems success model contains a six factors: system quality, information quality, system use, user satisfaction, the effect on the individual user, and the effect on the organization.

The paper [10] introduces and formulates the TourMustSee problem and proposes linear programming approach with must-see POIs constraint for solving tourist trip planning problem. This problem is similar this Orienteering problem where the objective is maximize a total score for tourist. Dataset of POIs is taken from Yahoo Flickr Dataset and every POI has own utility score for tourist. The authors compares 4 different algorithms based on Integer Linear Programming approach with different metrics such as inclusion rate of must-see POIS, route profit and utilized time budget. The comparison shows that their solution performs consistently better compared with various baselines

The authors of the paper [9] are develop a personalized travel planning system which uses the time framework concept with regulated recommendations to users. The travel planning module create a preliminary travel schedule by using a schedule determination method. This method includes distance, popularity and time calculations. The information about all matching locations and time block categories are treating as input parameters for the schedule determination method. For selection the destination point, which fits the tourist requirements, the travel requirement module cross-joins user requirements and the travel destination. Places that meet the user's requirements are connected and formatted in time framework depending on the time for planning a preliminary trip schedule.

The authors of the paper [12] uses information from Location Based Social Networks such as FourSquare for tourist attraction recommendations constructing. Each recommendations is a sequence of multiply points of interest with some spatio-temporal constraints. The set of recommendation forms package with several factors as popularity of POI, distance between POIs, etc. Graph-based solution is taken for route construction. At first, attraction selection is performed by using two-leveled filter (filter by distance around tourist and filter by authors defined features such as user preferences for attraction type or attraction popularity). Second, the probability of visiting edges between selected attraction nodes based on global sequence patterns and travel time between nodes is computed. Finally, the authors use Floyd-Warshall algorithm to find all shortest parts between attractions.

In the paper [11] the authors represent the PlanTour system, which creates personalized tourist routes using the information collected from the travellers social network MINUBE. The system uses an automated planning approach to create a multi-day plan with the most interesting sights of the visited city/region. The developed system collects information about users and points of interest from the MINUBE, groups these points using clustering methods to divide the problem into subtasks. After these steps the system uses an independent from the tourist area automated scheduler, which finds tourist plans. Unlike other tourist recommendation systems, the PlanTour planner can organize the relevant objects of interest, taking into account the expected travel of the user and user estimates from a real social network.

In the paper [14] the authors proposes the Personalized Crowd-aware Trip Recommendation algorithm for generating personalized trips with avoiding the most crowded POIs. Tourist preferences is extracted by using collaborative filtering. The modification of the Ant Colony optimization is used for attraction tour generation, which can merge tourist interests with POI popularity and crowdedness. The algorithm evaluation was done by using traffic information obtained from real-life pedestrian sensor dataset and tourist travel experience extracted from Flickr photo dataset.

The paper [13] presents the eCOMPASS multimodal tourist tour planner. This tourist tour planner provides a navigational aid and incorporates multimodality within its routing logic aiming at deriving near-optimal sequencing of points of interest (POIs). Generated recommended tours by this planner tries to achieve optimal minimization waiting time at transit stops. The eCOMPASS allows users to define arbitrary start/end locations (e.g. the current location of a mobile user) rather than choosing among a fixed set of locations. The core engine of eCOMPASS is based on a novel cluster-based heuristic approach, the SlackRoutes. The main idea behind approach is to motivate tourists to visit areas with high density of "good" candidate vertices. The SlackRoutes takes into account time dependency while calculating travel times from one vertex (i.e. POI) to another.

The authors [7], [9], [11], [12] don't take into account context situation and other authors ([8], [10], [13], [14]) works with context partially. The system proposed by these authors would work with traffic jams situation, weather situation, attraction popularity and work-time restrictions while tourist trip constructing. The ontology based approach allows to handle with knowledge semantic heterogeneity and describes tourist and context information. Each service in the proposed system works with proposed ontology.

III. Tourist trip planning system

The proposed tourist trip planning system is a service-oriented system which provide attraction visiting route and attraction information to a tourist. This system offers to use a context-oriented approach for constructing tourist attraction visiting route in his/her region. Context situation in the region and user profile are formalized accordingly to an ontology. The system can store a region information in sqlite-database and provides it to user for offline attraction processing on a smartphone. For building the attraction visiting route, methods and algorithms of graph theory are used.

A. Use case

The tourist leaves for an unfamiliar country and has a certain schedule (e.g. he/she leaves the hotel at 11 am, and his/her departure from the country — at 20 pm). The proposed tourist trip planning system analyzes the proposed timetable, takes into account the time restrictions/user wishes and offers a tourist trip to local attractions. While compiling a tourist trip, the system must take into account the user personal preferences (either pre-defined or calculated using the collaborative filtering technique) and the current situation in the city (season, weather, traffic jams, etc.).

Access to the system is provided by using a personalized electronic devices (smartphones, tablets, etc.). They can be used during the trip to provide information about the surrounding attractions, through audio and video signals based on the current location of the tourist and its speed. An attraction information is based on combining various information parts (pre-recorded speech of the tourist guide, images/video, etc.)

on the fly in a personalized manner taking into account the context.

By using smartphone, the tourist can also view a current map of the city with nearest attractions. During a trip, the user will get a proactive recommendations based his/her location, orientation relative to attractions and another context information. In case the user is not willing to use the Internet to get information about the map of the region and information about attractions (e.g. finding a tourist in roaming), the tourist trip planning system needs to provide cached data about the city and the region.

B. Reference model

A reference model of tourist trip planning system is presented on Fig. 1. This system provides content to an user and consists of a set of interacting services. The user uses a smartphone/tablet for viewing attractions on current region map and information about them and interact with services for accessing to a constructed attraction route. The device stores a user profile, which includes information about the user and his/her preferences.

The *recommendation service* will use user information, his preferences and ratings to attraction and media files for more accurate individual recommendations. The service takes tourist ratings associated with each attraction by all tourists as an input and performs user-based collaborative filtering.

The *content delivery service* stores all data about cities and related attractions. This kind of information can be distributed in two ways. First way is providing partial information online on the user's request — e.g. service gives information about the specific attractions. Second way is providing whole data array about city and related attractions to user, because he want to use it in offline mode in future.

The *information aggregation service* uses OpenStreetMaps data for gathering geo-information about city and related attractions. Next, the service tries to match the collected attraction information with information, which is taken from Wikipedia project. All matched information is stored into a database.

The *context service* is designed for storing and working with a context information. The context has been split into two parts: user context and region context. User context contains information about location and movement of user; user time restrictions (e.g. tourist arrived to a region at the morning and have free-time until evening). Region context contains information about the current traffic situation; attractions schedule of region and information about current weather and season. Traffic and weather information can be obtained via appropriate services — *weather service* and *traffic service*.

The *route planning and creation service* is interacting with *recommendation service*, *context service* and *content delivery service* for creating an attraction route for the user. The purpose of the context-oriented tourist trip planning service development is to increase the tourist attractiveness and total saving of the tourist time when visiting the attraction.

More detailed information about the tourist trip planning system conceptual model can be found in paper [16]. In the current step of reference model development each service works with ontology in section III-C which leads to semantic interoperability of services. Each service need to work with own local ontology which is designed according to the proposed ontology. In case of service interacting, services operate with already predefined concepts and doesn't need to match ontologies. This approach with shared ontology allows to accelerate and simplify the whole development process of tourist trip planning system.

C. Ontology

The current iteration of a proposed ontology is presented on Fig. 2. This ontology can be divided into two parts. The first part describes a tourist, his/her intentions and environment around him. The second part shows the context information which affects route construction. OWL language is chosen for the ontology description. This language can describe classes and relationship between them. This language is much expressive in terms of syntax and give more possibilities for ontology creation compared to other ontology languages.

A tourist is described by classes: `User`, `Profile` and by the subclass `Preferences` (Fig. 3). The `Profile` subclass contains following data properties: *id* — identification number for a tourist in the developing system, *name* and *surname* defines person itself and *age* can be used for recommendation system as one of possible parameters. The `Preferences` subclass represents the tourist preferable attraction types. All personal tourist information can be filled manually (e.g. on a smartphone application startup) or automatically based on pre-existed profile in other systems (e.g. Google of Facebook account). User interacts with services (class`Service`) for creating desired attraction route. An estimated tourist route is placed in the class `Attraction_route`, which connected with `User` by the object property *wants*. This class contains information about constructed attraction route and stores *efficiency* and *time_limit* data properties as user route rating and route time duration accordingly.

The tourist is located in city, which is described by class `City`. Every city in this ontology contains general information — *title* and *coordinates* of geographical city center. The city can be divided into district (subclass `District`) with some information (*district_title* and boundaries (*district_boundaries*)). Attractions are described by class `Attraction`, located in tourist city and fill tourist route. This class contains general information (*attraction_title*, *attraction_description* and *attraction* data properties), tourist *attraction_rating* (can be used in further recommendations) and type (e.g. museums, parks, theatres, historical places, e.t.). Media information (subclass `Attraction_media`) can be attached to attraction and contains *media_title*, *media_type* (Text-To-Speech, audio/video) and *media_duration* data properties.

A process of the route construction can be affected by such factors as a tourist route restrictions (class `Route_restriction`) and current contextual situation (class `Context`) (Fig. 4). The route restrictions are formed by the user and contains the following data properties: *route_endpoints* includes start and finish points specified by the tourist, *time_restrictions* contains information about expected

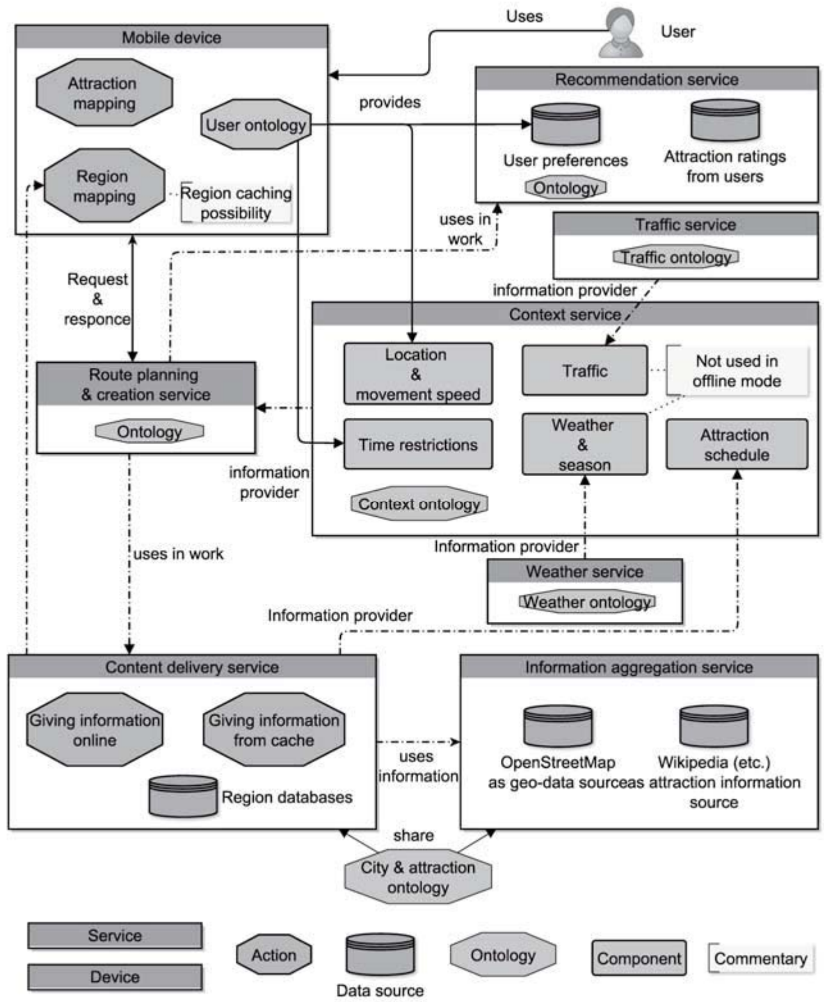

Fig. 1. Toursit trip planning reference model

trip duration, *desired_attraction_types* describes allowed by the tourist attraction types (e.g. the tourist prefers museums and monuments but doesn't want to visit castles and parks) and `tourist_movement_restrictions` contains a current way of the tourist travel (the tourist travels by foot or by car).

Contextual situation can be divided into two parts: tourist context (subclass `User_context`) and tourist city context (subclass `City_context`). User context contains information about current location (subclass `Position`) with the following properties: *coordinates*, *speed*, *movement_type* (car movement or by foot) and *direction* (can be used for more precised proactive recommendations). City context can be described by traffic jams (subclass `Traffic` with overall jams score on whole city), attraction information (subclass `Attraction_context`) and current weather situation (subclass `Weather`). The subclass `Weather` holds information about current *season*, *temperature*, *pressure*, *humidity* and wind information based on subclass `Wind`. The subclass `Attraction_context` contains *time_restriction* information about opening and closing hours for attraction, place *popularity*, which can be used in overall attraction rating and information about attraction location and type (*open_area*). The class `Time` and the subclasses `Interval` and `Instant` can be used for state identification on the time line.

All services mentioned in conceptual model (section III-B) store information accordingly to the proposed ontology. Each service has own smart space — the service can interact with restricted ontology part only. The *recommendation service* fills the class `Preferences` and works with attraction ratings from `Attraction` class of ontology. The *context service* stores data accordingly `Context` and `Route_restriction` classes. The tourist location information is taken from a smartphone sensors and stored in `User_context` and `Position` classes. Traffic and weather information are gathered from related services and the *context service* puts this kind information to `Traffic` and `Weather` classes accordingly. The *information aggregation service* gathers all information about city (class `City`) and related attractions (`Attraction` and `Attraction_media` classes). The tourist can fill the class `Route_restriction` by using the smartphone and the *Route planning service* gathers all information and writes constructed attraction route into `Attraction_route` class. In case of services interaction between each other, the service need to attach to request a user/city/attraction link. This modular approach allows fast service developing with accurate storage of information according

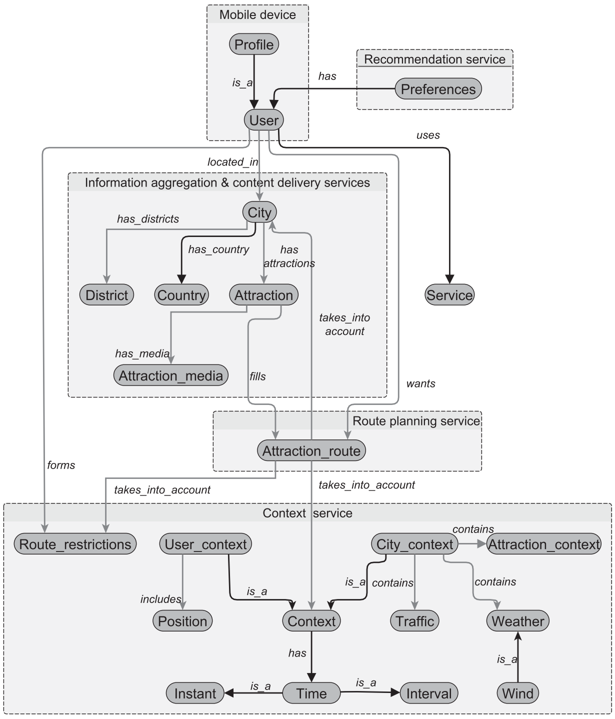

Fig. 2. Ontology for the tourist trip planning system

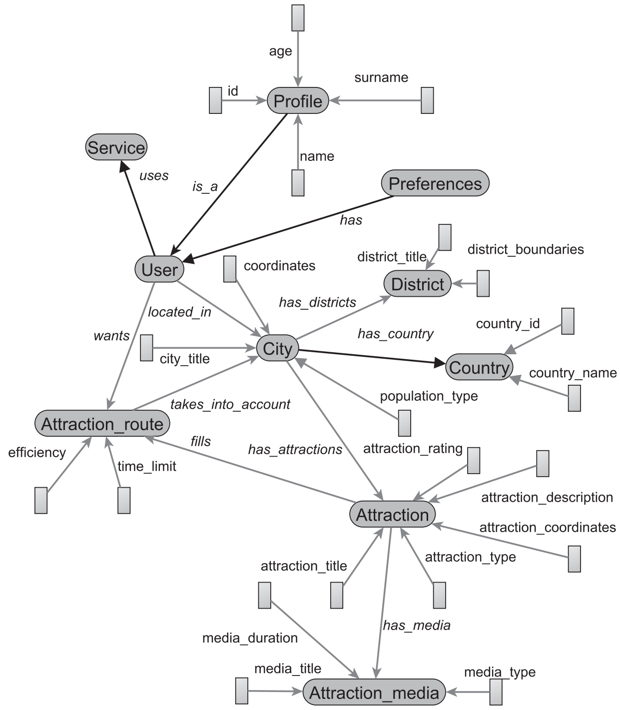

Fig. 3. User ontology

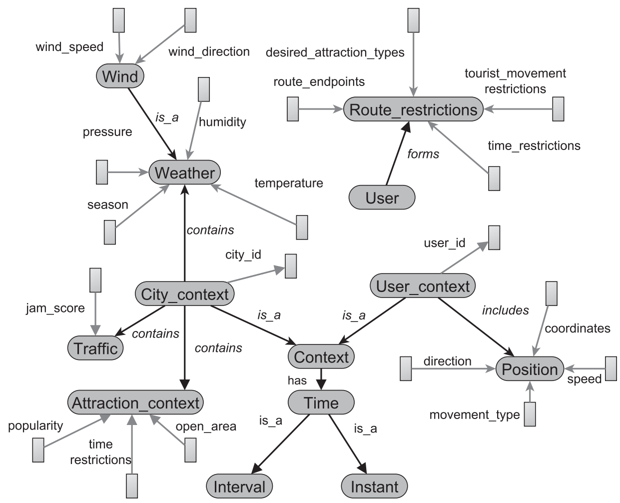

Fig. 4. Context ontology

to the ontology.

The proposed ontology can be easily modified and extended and can work with upper-level ontologies. For ontology building authors used the free, open-source ontology editor Protègè. This editor offers a lot of tools for ontology creation and manipulation. However, the ontology visualization was made in the proprietary cross-platform graphical editor yEd with a great choice of graph visualization tools. The yEd editor stores constructed schemes in an XML-based GraphML format. It is possible to use custom XSLT stylesheet to transform a OWL ontology to a GraphML format.

IV. Implementation

The Android-base prototype implementation is based on Tourist Assistant — TAIS [15] mobile application. At the current development progress the prototype has the following functionalities: display a list of available regions; download the region database for further work offline (attraction information stores at the user device); provide attraction list with image gallery for current position, show nearby attraction in interactive map (Fig. 5) and provide a proactive recommendations while give proactive advice while a tourist visits the city. The *information aggregation service* needs to work before system launch, because this service is designed to gather attraction information in offline phase. The *content delivery service* proposes to user the offline cache with region data include attraction information with media files. The *context service* keeps track on user context information and weather situation. The more detailed information about services implementation can be founded in paper [16].

The *weather service* can be divided into two parts: main Python-based service, which gathers weather information and retrive it on user/service request and local gecoder-service based on Postgres database and Postgis extension. The weather data stores in a Redis storage accordingly to the proposed ontology. Main service caches a request to several weather data providers such as OpenWeatherMap and keeps data by a certain amount of time. This action allows the weather service requests not to overload the weather data providers in case that multiple requests coming from the same city weather. In case of using multiply weather data providers the forecast with best accuracy will be stored or, in case if providers doesn't provide this kind of information, average forecast will be used. Local geocoder service is used for reverse geocoding task — tourist city determination by its coordinate. This method allows

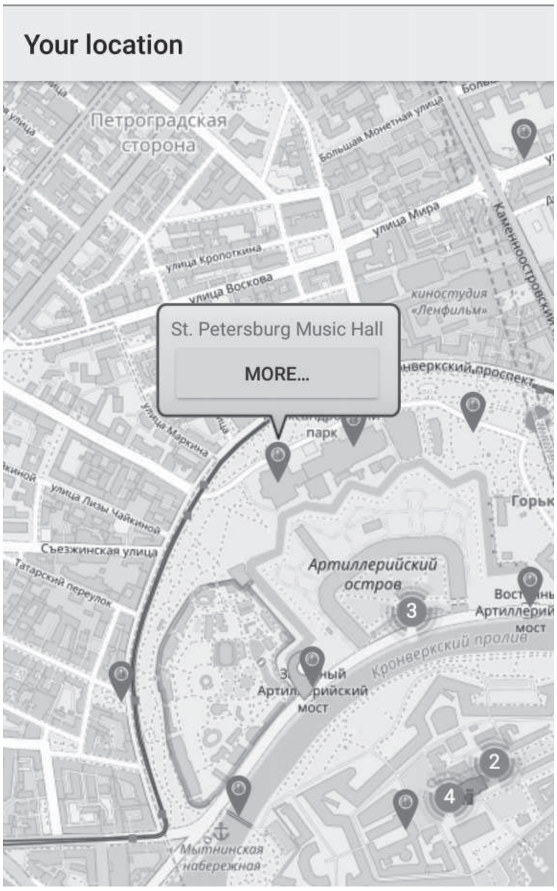

Fig. 5. Prototype implementation

to optimize the performance of the entire weather service by reducing requests to third-party geocoder services.

The services interaction was performed by using the concept of smart space. Smart space aims to seamless integration of different devices by developing ubiquitous computing environments, where different participants can share information with each other, make various computations and interact for joint task solving. The open source Smart-M3 platform allows to perform access to ontology oriented information and knowledge. Smart-M3 contains kernel and knowledge processors (KP). Kernel consists of two parts: semantic information broker (SIB) and information storage. SIB receives incoming requests for cooperation manipulations with the information storage from knowledge processor and sends back operation results. There are different implementations of semantic information broker, provided by different members of the community [6]. All information from the information storage is kept as a graph that is formed according of the rules of structured data representation RDF — Resource Description Framework. Information in the graph is described as a triplet — "subject — predicate — object" (a subject uses a predicate to affect on an object). The subject and the predicate can be URI (Uniform Resource Identifier), the object can be URI link as well as literal (some value with certain type). Knowledge processors function on the smart space device software. Knowledge processors interact with SIB using SSAP protocol — Smart Space Access Protocol. SSAP protocol operations are described in XML format.

The general scheme of service interaction is presented on Fig. 6. Every service stores information into Smart-M3 platform according local ontologies which are based on proposed ontology in section III-C. The *context service* needs to update weather information for a specific user. This service sends to the Smart-M3 platform the following triplets:

- weather request triple — [*"<user_id>"*, *"http://cais.iias.spb.su/RDF/tais#weather_request"*, *"True"*], where *"<user_id>"* is a tourist identification number (in case of Android prototype — unique Google Mail account)
- user location ([*"<user_id>"*, *"http://www.w3.org/2003/01/geo/wgs84_pos#lat"*, *Float*], [*"<user_id>"*, *"http://www.w3.org/2003/01/geo/wgs84_pos#lon"*, *Float*])

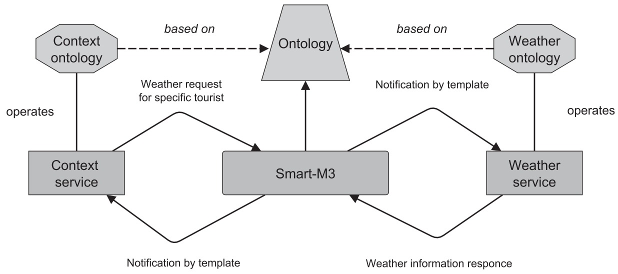

Fig. 6. Service interaction based on the Smart-M3 platform

The SIB accepts the received request, loads the related triplet information in the information storage and notifies services about triplet inserting by using the "publish-subscribe" mechanism. The *weather service* subscribes to all weather requests and identifies the tourist city based on coordinates by using reverse geocoding. In case of successfully city identification, the service checks current weather state and its time relevance. If weather state is outdated the weather service requests new information from multiply providers and chooses the most accurate forecast for selected city, otherwise the service retrieves weather information from cache.

After getting current weather situation, the service forms a list of triples through the following template: [*"<user_id>"*, *"http://cais.iias.spb.su/tais/<type>"*, *<value>*], where *"<user_id>"* is a tourist identification number, *"<type>"* is a certain weather predicate and *"<value>"* is a subject value. The possible values for *"<type>"* are: *weather_icon*, *weather*, *temperature*, *wind_speed*, *wind_direction*, *humidity* and *pressure*. The created triplet list is transferred to the Smart-M3 platform and the *context service* gets response.

V. Conclusion

This paper introduces the ontology tourist trip planning services which allows interoperability between each other. This ontology covers tourist, his profile and preferences, city and its attraction, tourist route and context. Services works in "smart space" — the service can interact with restricted ontology part only. Usage of proposed ontology allows to skip ontology matching between interacting services. The case of the weather service interaction with context service was presented and described.

Tourist trip planning services are presented on the reference model. The personalized attraction trip route is constructed by the interaction of services. The context situation and users preferences are taken into account when the system constructs the route. As the future work the authors are planning to implement route constructing service with context and user preferences usage and evaluate the proposed ontology.

ACKNOWLEDGEMENT

The presented results are part of the research carried out within the project funded by grants # 18-37-00337, 17-29-03284, 16-29-04349 of the Russian Foundation for Basic Research and Project # 618268 by ITMO University.

REFERENCES

[1] Zheng Xiang, "From digitization to the age of acceleration: On information technology and tourism", *Tourism Management Perspectives*, vol. 25, 2018, pp. 147–150.

[2] Marchenkov S. A., Vdovenko A. S. and Korzun D. G., "Enhancing the opportunities of collaborative work in an intelligent room using e-tourism service", *SPIIRAS Proceedings*, vol. 50, 2017, pp 165–189.

[3] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas and Grammati Pantziou, "Mobile recommender systems in tourism", *Journal of Network and Computer Applications*, vol. 39, 2014, pp. 319–333

[4] Smirnov Alexander, Kashevnik Alexey and Ponomarev Andrew, "Context-based Infomobility System for Cultural Heritage Recommendation: Tourist Assistant–TAIS", *Personal Ubiquitous Comput.*, vol. 21, n. 2, 2017, pp. 297–311.

[5] Smirnov Alexander, Ponomarev Andrew, Shilov Nikolai, Kashevnik Alexey and Teslya Nikolai, "Ontology-Based Human-Computer Cloud for Decision Support: Architecture and Applications in Tourism", *Int. J. Embed. Real-Time Commun. Syst.*, vol. 9, num. 1, 2018, pp. 1–19.

[6] F. Viola, A. D'Elia, D. Korzun, I. Galov, A. Kashevnik and S. Balandin, "The M3 Architecture for Smart Spaces: Overview of Semantic Information Broker Implementations". *Proceedings of the 19th Conference Open Innovations Association FRUCT*, Jyväskylä, Finland, November 7–11, pp. 264–272, 2016.

[7] Chia-Chen Chen and Jia-Lun Tsai, "Determinants of behavioral intention to use the Personalized Location-based Mobile Tourism Application: An empirical study by integrating TAM with ISSM", *Future Generation Computer Systems*, 2017.

[8] Gavalas Damianos, Kasapakis Vlasios, Konstantopoulos Charalampos, Pantziou Grammati and Vathis Nikolaos, "Scenic route planning for tourists", *Personal and Ubiquitous Computing*, vol. 21, 2017, pp. 137–155.

[9] Hsiu-Sen Chiang and Tien-Chi Huang, "User-adapted travel planning system for personalized schedule recommendation", *Information Fusion*, vol. 21, 2015, pp. 3–17.

[10] Taylor Kendall, Lim Kwan Hui and Chan Jeffrey, "Travel Itinerary Recommendations with Must-see Points-of-Interest", *Companion Proceedings of the The Web Conference 2018*, 2018, pp. 1198–1205.

[11] Isabel Cenamor and Toms de la Rosa and Sergio Nez and Daniel Borrajo, "Planning for tourism routes using social networks", *Expert Systems with Applications*, vol. 69, 2017, pp. 1–9.

[12] Hti Rashmi and Desarkar Maunendra Sankar, "Personalized Tourist Package Recommendation Using Graph Based Approach", *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*, 2018, pp. 257–262.

[13] Damianos Gavalas, Vlasios Kasapakis, Charalampos Konstantopoulos, Grammati Pantziou, Nikolaos Vathis and Christos Zaroliagis, "The eCOMPASS multimodal tourist tour planner", *Expert Systems with Applications*, vol.42, num. 21, 2015, pp. 7303–7316.

[14] Wang Xiaoting, Leckie Christopher, Chan Jeffrey, Lim Kwan Hui and Vaithianathan Tharshan, "Improving Personalized Trip Recommendation by Avoiding Crowds", *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 25–34.

[15] Smirnov Alexander, Kashevnik Alexey, Ponomarev Andrew, Teslya Nikolay, Shchekotov Maksim and Balandin Sergey, "Smart Space-Based Tourist Recommendation System", *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, 2014, pp. 40–51.

[16] Mikhailov Sergei and Kashevnik Alexey, "Smartphone-based tourist trip planning system: a context-based approach to offline attraction recommendation", *MATEC Web Conference*, vol. 161, 2018, pp. 03026.