An Approach towards Real-Time Data Exchange Platform System Architecture

Bernd Resch, Francesco Calabrese, *Member, IEEE,* Assaf Biderman, *Member, IEEE*, and Carlo Ratti

***Abstract*—Recent developments in the technology industry and academia show that users are increasingly becoming the focus in technology development. Not only new applications capable of performing certain tasks are brought to potential users, but also technical innovations that are built around people in order to best fit their needs.**

Beyond standard web and desktop applications, an ideal use of this principle would be in a digital representation of a real-world city which allows city inhabitants to access, add, and modify information and thus use the city in a more informed way. The term WikiCity represents this concept.

The development of a WikiCity requires answers to questions such as: how can a city be sensed, modeled, and actuated in real-time? How can inhabitants interact with information in the city?

The paper presents this idea through a use-case scenario in which users can inform themselves about certain events in the city. They can retrieve real-time information related to the events such as nearby happenings, availability of public transportation, and traffic conditions, or create a personalized event route. Users can also feed information back into the WikiCity system before, during, and after attending events.

This requires new developments in modeling of human behavior, flexible back-end infrastructures, and semantic-web. The architecture of such system including the relations between data sources and user services, and data flow procedures are explained in the paper. A first implementation of the WikiCity concept was presented at the White Night event in Rome, Italy on September 8, 2007.

***Index Terms*—digital city, ontology, real-time city, semantic web, real time control system.**

I. Introduction

HOW can a city perform as a real-time control system? An answer to this question offers new possibilities for more efficient usage of city resources, social interaction, emergency response, and other commercial applications for the urban area.

The WikiCity concept, builds on previous concepts presented in the Real-Time Rome project [1] with the aim of creating a real-time data exchange platform in which city inhabitants can participate. Aside from other applications, such system could enhance the perception of the city as an infrastructure for participation, social inclusion, and collaboration.

Real-time representation of a real-world city in digital form requires research in several scientific domains. For example: modeling of human behavior (social interactions, decision-making, or strategy) and, flexible back-end infrastructures, semantic-web ontologies for the appropriate knowledge domains represented in the system, and fusion of real-time and archived data from diverse heterogeneous sources.

Practical applications of WikiCity include real-time, location-based, and temporally up-to-date information systems (e.g. upcoming events in the vicinity, nearest emergency facilities, or route planning functionalities), real-time traffic monitoring and management system (e.g. for disaster relief or transportation management), or information broadcast on public spaces (e.g. e-government messaging via large screens).

This paper presents the conceptual background behind WikiCity and proposes approaches to its implementation. Chapter II offers comparison between WikiCity and related work; chapter III discusses challenges, and chapter IV demonstrates a use-case scenario. Chapter V contains a system description from a technical viewpoint including; chapter VI describes a topological view of the WikiCity system architecture. In chapter VII, different data flow procedures are explained through examples from the use-case scenario to give a better understanding of the system architecture. Chapter VIII presents a comprehensive implementation scheme, showing how heterogeneous services are integrated through a process choreography scheme. This is exemplified through a preliminary demonstration of the project which was shown during the White Night in Rome, Italy, on September 8, 2007. The paper concludes with a short summary and a future outlook.

II. Related Work

In past research efforts, the term 'city' has been used increasingly in reference to a real-world space which offers social opportunities and cultural landmarks for urban analysis [2]. Yet, so far, implementations have been based on ideal users whose characteristics were selected and simplified for specific applications. Recently, social software service implementations try to promote different user-specific interpretations of urban landscapes and by that enable the development of multiple meanings and representations of city-use. In [3], a current paradigm shift in urban computing is discussed. Rather than merely reflecting the city, the city itself and its inhabitants become part of a computer application, and urban computing becomes part of the city. As a consequence the city develops into a platform for the collective creation of content and social interaction, enhancing the perception of the city as a source of experience. The authors mention disconnection, dislocation, and disruption as the three main challenges when designing urban applications.

Current trends in ubiquitous computing suggest integrating applications into everyday life, allowing users to interact with them while performing ordinary activities rather than being required to engage in a dedicated interaction procedure.

Developments in pervasive information involve smooth integration of user services with social activity, fast and personalized services, and seamless sharing of information between services through multiple types of interfaces [4].

III. CHALLENGES OF THE WIKICITY CONCEPT

One fundamental challenge is in using very large amounts of user-generated data to provide useful information to someone who is moving through physical space without causing information overflow. This requires an investigation into how mobile input devices and physical spaces differ in terms of the action possibilities they offer to the user and understanding how they can complement each other [5].

Amongst the important practical issues in implementing concepts of digital urban landscape are: transfer bandwidth, data freshness, data propagation latency, application-relevant metrics, handling of big amounts of data (aggregation or prioritization), heterogeneity in structure, topology, data ownership, and data dissemination quality, as described in [6]. The authors also mention that fulfilling basic requirements for mobile applications in urban environments such as data persistence, coordinated deployment, respect of users' privacy, connectivity, or compatibility, is an essential but in no case trivial problem. Further implementation challenges of mobile services include real-time computing, location information, human-system communication, routing algorithms and real-world data handling [5].

Another important matter to consider in the design and deployment of mobile applications in an urban context is that people might use a certain service in unexpected patterns. Hence, the system's back-end must be designed flexibly while the user interface should be intuitive.

Finally, a crucial requirement for the WikiCity system to function is a comprehensive and domain specific semantic-web ontology. In [7], the author describes an ontology as an explicit specification of the conceptualization of a domain coupled with a specific vocabulary and a set of assumptions concerning the meaning of the vocabulary objects. This seems to make a formal determination of an ontology's structure hard if not impossible. Thus, recent trends in ontology creation increasingly engage the users in the process [8].

The approach proposed here and the integration of different research areas, as mentioned above, becomes important to the development of location-enhanced applications, and the growing requirement for appropriate development toolkits and supporting platforms [9].

IV. WIKICITY CONCEPT – USE CASE

This paper proposes a system architecture for a real-time data exchange platform. The different components within it are described through a use-case scenario to illustrate the topological scheme, implementation concept, and data flow.

Imagine a series of events, such as a festival, that are taking place in a large city and distributed over a large area. You are interested in attending some of them and, as a first step, you decide to learn where and when are events taking place. You query a news-feed service of a public newspaper which also includes feedback from previous visitors. Having chosen several events, you retrieve a travel route from one to the next. Before leaving for the festival, you check traffic conditions for the best route given current congestion levels and parking availability.

Following the first event, you would like to share your experience with other users of the system. Using your mobile phone, you upload your feedback, making it instantaneously available within the system.

As you proceed to the next event, you look into the availability of public transportation. The positions of all relevant vehicles in your proximity are illustrated clearly on a map for your selection through a public screen.

In case you would like to change your plan on-the-fly, your route can be adapted dynamically and events can be added based on various methods of filtration.

The system can also help you find additional services in your vicinity (from hotels and dining, through emergency facilities, to custom services defined by other users).

Fig. 1 illustrates the relationship between different data sources and services used by the WikiCity system to provide the functionalities in the scenario above. Naturally, this requires a homogenization mechanism to make different kinds of data compatible.

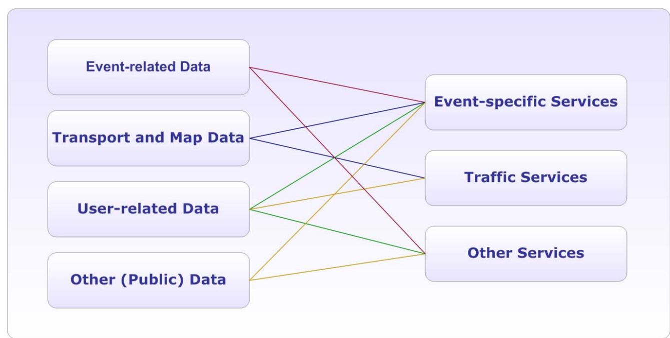

Fig. 1. Relation of Data and Services.

V. WIKICITY CONCEPT

This chapter provides a description of the conceptual and topological architecture of the WikiCity system, starting with a basic introduction of the technical concept (V.A), followed by a description of public interfaces used in the system (V.B).

A. Basic Configuration

As stated, the aim behind WikiCity is to create a digital representation of a real-world city. City inhabitants can provide real-time information to the system, and other service providers can also connect data streams via automated web services. The relationship between the real-world city and the WikiCity is mediated by user services, data, and processes.

User services mediate between the end-user and the WikiCity system by providing capabilities to perform various operations on the data in the system. Here is an example scenario referring to the use-case described in IV: an end-user plans a trip to a festival using a *user service* by executing a *process* which combines real-time information such as traffic conditions, availability of public transportation, the weather forecast, and event information. After having visited several events, the user provides feedback and thus updates the WikiCity system.

B. Public Interfaces

In general, the WikiCity system can be accessed through three different web interfaces, as illustrated in Fig. 2.

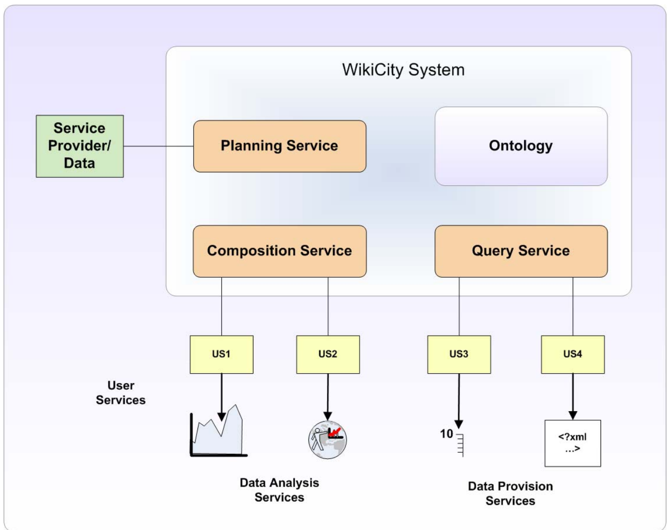

Fig. 2. Public Interfaces of the WikiCity Infrastructure.

The *Query Service* allows users, which can be either end-users or other web services, to query data from the system.

The *Composition Service* enables users to create a process by using an existing Semantic Web Services (SWS) and Processing Services (PS).

The *Planning Service* constitutes an interface for external service providers to register new SWS or PS to the WikiCity.

Besides these public interfaces, the system has three components that compose the data and metadata back-end. As it can be seen in Fig. 2, User Services are divided into Data Provision Services (DPS) and Data Analysis Services (DAS). DPS return data as retrieved from the WikiCity system without further processing, with an output format of XML, text, or any value of a variable in the database. The DAS provide pre-processed data. For example, data that can be presented on a map, excluding any type of analysis (such as GIS functions). It is the choice of the User Service provider, whether to offer a DPS or a DAS, and to decide how the data should be processed.

Examples for DPS are:

- an XML representation of a list of public means of transport in a certain area
- a single entry obtained from the database such as the location of a restaurant, or the contact details of a taxi company
- the temperature in a certain location or area

Examples for DAS are:

- presentation of real-time population data on a map
- graphical representation of a calculated fastest-route from one geographic location to another
- the interpolated temperature map for a certain area

VI. System Architecture

As the WikiCity system is intended to be as flexible as possible, and the back-end service chain includes a complex combination of different processes.

Generally speaking, the system infrastructure has to deal with three different procedures, data queries, service planning, and processing composition requests (as shown in Fig. 2). In addition to the public interfaces described in section V.B, the system also includes an ontology and two additional services that handle data flow within the system:

The *Ontology* is a data model that represents a set of concepts (categories) within a domain and the relationships between those concepts. The ontology is described by means of the Web Ontology Language (OWL). Generally, all data in the system are related to one or more categories in the ontology, which is the basis for a structured query system and well-defined data maintenance.

The *Ontology Service* is used to upload and access information about the data within the domain described by the ontology. It makes use of the *Ontology Service Interface* in order to query the ontology (e.g. using SPARQL). The *Ontology Service* is designed to:

- add new concepts (categories) to the ontology
- create connection between SWS, PS and the ontology

The *Proxy Service* is used to access data from the registered SWS and PS and to handle communication between the internal services and the semantic and processing web services. It makes use of the *Service Registry*, which contains a description of the services (WSDL files) and of the connections between them and the ontology (OWL-S).

The complete system with all components mentioned above including their interrelations is illustrated in Fig. 3. This figure also indicates the two different types of services that can be added to the system in order to retrieve information or process data for the purpose of combining several data sources in a pre-defined way.

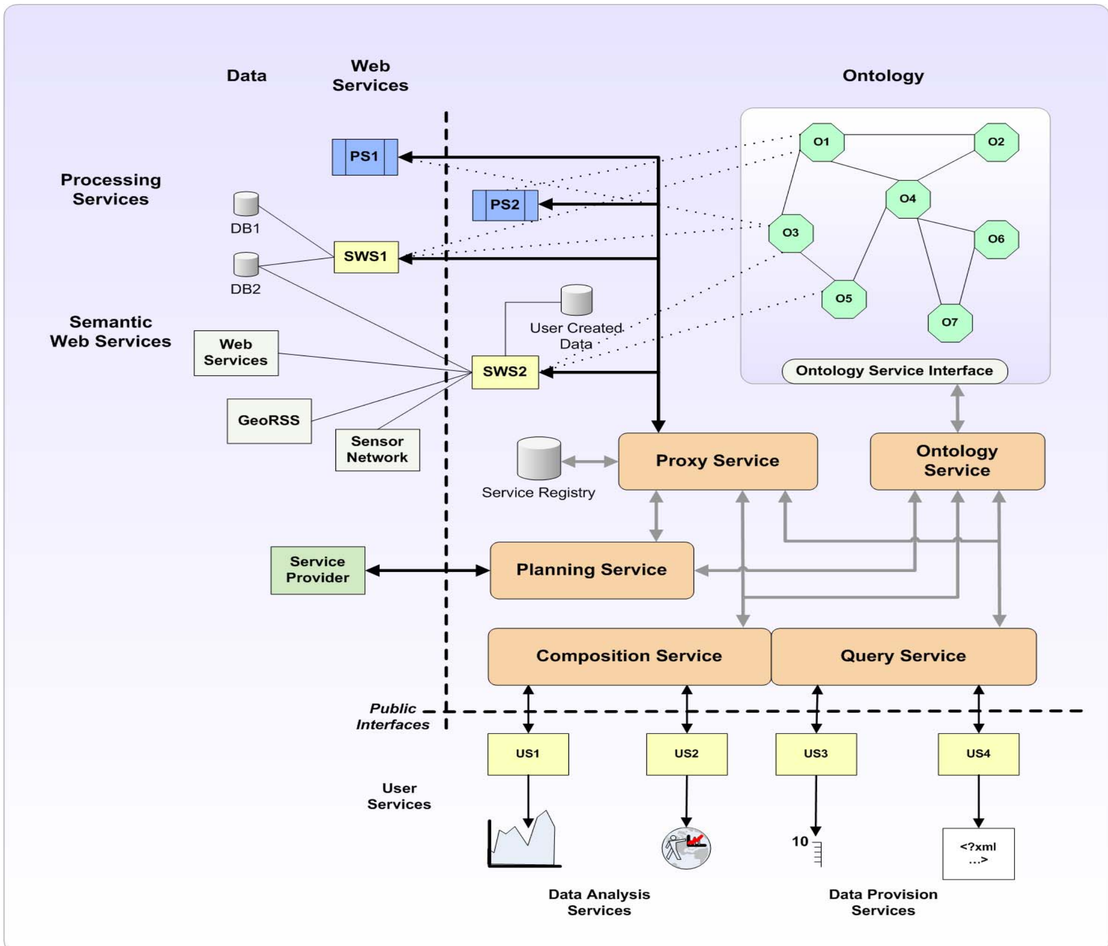

Fig. 3. Public Interfaces of the WikiCity Infrastructure.

Semantic Web Services (SWS) provide an interface between data providers and the WikiCity System. Data providers push their data to the SWS via web their services and also provide a semantic description about it to the SWS.

Processing Services (PS) are web services that can be used to process data within the domain described by the ontology (e.g. data merging and optimization algorithms).

Both, SWS and PS can either be offered by external service providers or hosted internally by the WikiCity system operator. External data feeds can be easily added to the system by connecting them to a SWS which maps the incoming data structure to the generic format used in the WikiCity system. On the other hand, single-user entries are stored in a central database. Although – from a technical viewpoint – single user-originated datasets are similar to data feeds, connecting them to an external web service makes sense only if large amounts of data are provided. Minimizing internet connections and easing maintainability by keeping a central database proves more efficient in this case.

The WikiCity system handles three different procedures: data queries, service planning, and processing composition requests. The following services handle these processes:

1) Query Service

The Query Service manages requests made by user services, in particular Data Provision Services, which can query data from Semantic Web Services (Fig. 3). Within the WikiCity system, the Query Service is connected to the Ontology Service for retrieving categories related to data objects, and to the Proxy Service for gathering service descriptions from the registry and relevant data from different SWS.

2) Composition Service

Composition Services manage two kinds of requests made by Data Analysis User Services (Fig. 3). The Composition can retrieve a business choreography from a Processing Service according to a request made by a Data Analysis Service. This choreography can be exectued by retrieving the relevant data, sending a request to a Processing Service, and returning the processed data back to the origniating User Service.

3) Planning Service

Planning Services manage planning and integration of new Processing Services as well as Semantic and Non-semantic Web Services (Fig. 3). An external Service Provider sends a service-planning request to the Planning Service, which queries the ontology for retrieving related categories, requests a corresponding list from the Service Registry through the

Proxy Service, creates the service, and then updates the Ontology and the Service Registry.

VII. Data Flow Procedures

This chapter describes different data flow processes to explain in detail how the services shown in Fig. 3 interact.

A. Data Requesting Procedure

Fig. 4 shows the data flow of a data requesting procedure. First, the User Service issues a request for all relevant categories in the ontology. It is passed from the Query Service to the Ontology Service, which queries the ontology. After having received a list of categories, the user selects a single category and receives a list of additional categories which are related to it (the type of relationship is pre-defined by the user). Based on this list, a request for the description of all SWS that offer data in at least one of the categories is sent to the Query Service and passed on to the Proxy Service. This procedure can be repeated to get SWS descriptions for any number of categories.

Once all desired SWS descriptions have been gathered, a data request can be issued by the User Service. The Query Service passes the request directly to the SWS, which returns the relevant data.

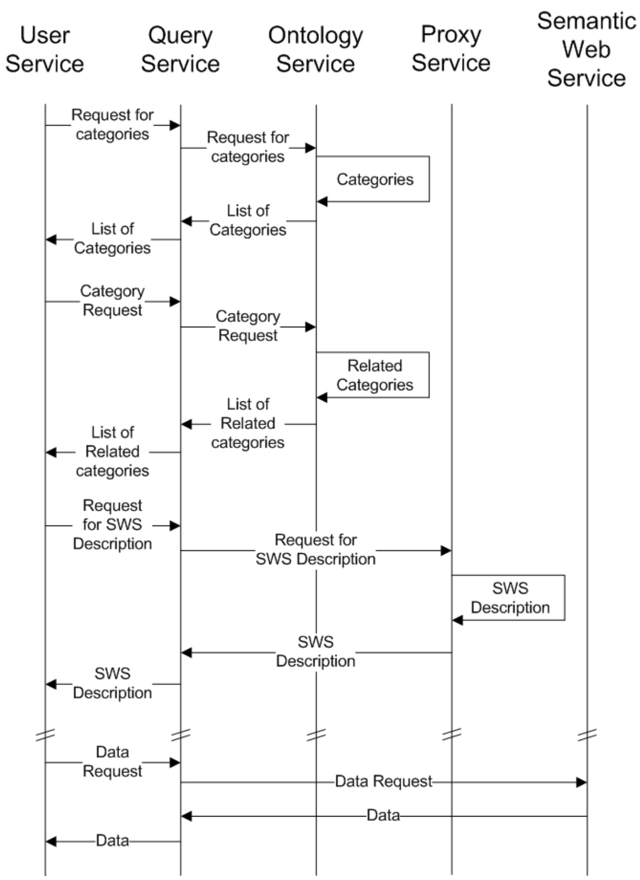

Fig. 4. Data Requesting Procedure.

B. Processing Services Requesting Procedure

In reference to the use-case described in chapter IV, this procedure (Fig. 5) combines real-time traffic information with information about current availability of parking spaces for a given destination, in order to plan the optimal travel route to an event. This procedure returns only the choreography for data processing. Sub-section VII.C describes how this choreography can be executed by a User Service.

The result of the Data Requesting Procedure described in VII.A is sent to the User Service, which sends a request for a list of Processing Services that can operate on the data categories retrieved in VII.A. The request is passed from the Composition Service to the Ontology Service, which queries the ontology for Processing Services related to the selected category. Based on the list returned, a request for a description of each PS is sent to the Proxy Service, which queries the Service Registry and returns the descriptions. The Composition Service then passes back to the User Service a list of all PS that can be applied to the data retrieved. Next, the User Service issues a processing request to the Composition Service, which creates a choreography for a process that combines different SWS with data sources and PS procedures. The last step can be repeated in order to update the SWS/PS choreography by adding new data sources or processing mechanisms.

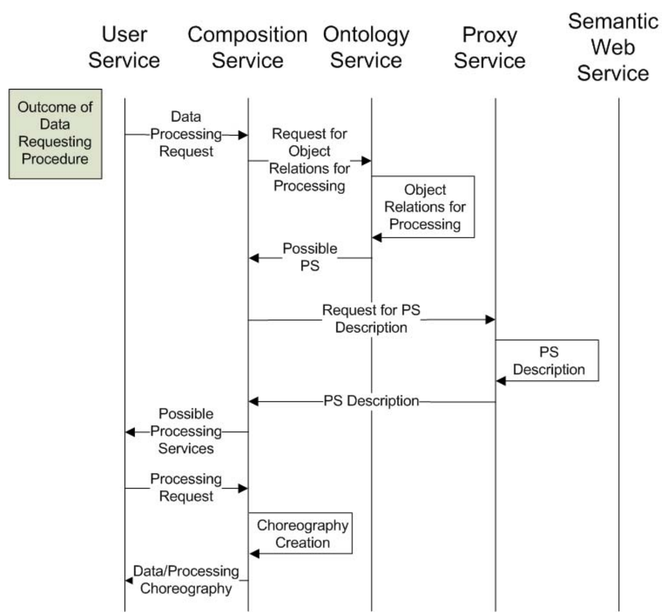

Fig. 5. Processing Services Requesting Procedure.

C. Choreography Execution Procedure

To execute the data choreography described in section VII.B, the User Service sends a request to the Composition Service, which executes the appropriate SWS and PS to retrieve and process the data. The composition service then returns aggregated data.

D. Semantic Web Service Planning Procedure

As described in chapter V.B, the Planning Service allows external service providers to add new SWS to the WikiCity *system. Practically, a telecom provider could add a data-feed to make real-time cell phone traffic available to the system. The SWS planning procedure (see Fig. 6) is initiated by a request from a SWS provider to the Planning Service for a list of categories contained in the ontology. The Planning Service forwards the request to the Ontology Service, which returns a list of categories. The SWS provider sends a SWS registration request, based on the list, to the Planning Service which forwards it to the Proxy Service. If the service does not already exist in the system, it is recorded in the Service Registry. The ontology is then updated with the new categories and a confirmation of registering the new service is sent to the SWS provider.*

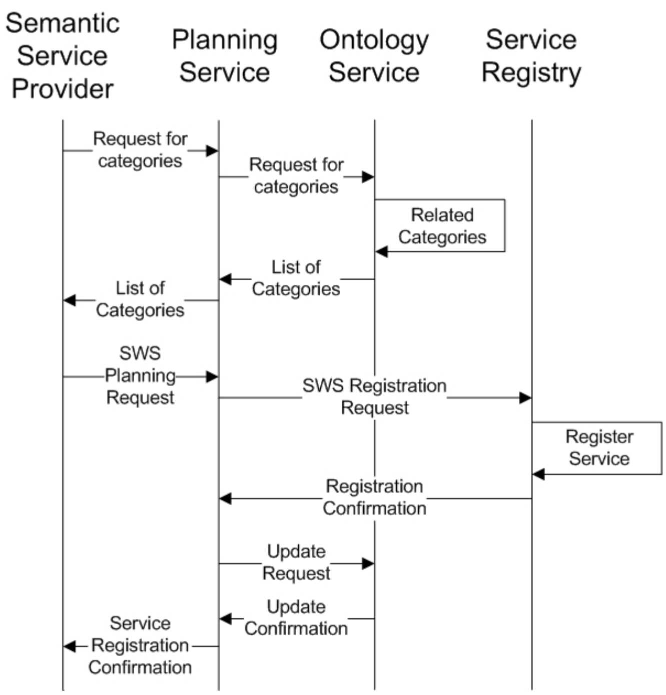

Fig. 6. Semantic Web Service Planning Procedure.

E. Processing Service Planning Procedure

This planning procedure is similar to procedure described in VII.D, with a slight modification. It is initiated by combined request from a PS provider to the Planning Service for a list of categories contained in the ontology and a list of registered PS. The Planning Service forwards the category request to the Ontology Service, which returns a list of categories, and the PS request to the Proxy Service, which returns a list of PS after having queried the Service Registry. The Planning Service then sends back the combined list of categories and PS to the processing service provider. Service registration is carried out as described in VII.D.

F. Non-Semantic Service Integration Procedure

Non-semantic services, such as web data sources, can also be connected the WikiCity system. The challenge here is mapping the incoming data structure to a SWS. The process starts with a request to the Planning Service for relevant categories registered in the ontology, including a list of SWS. The Planning Service forwards the category request to the Ontology Service, which returns a list of categories, and the SWS request is sent to the Proxy Service, which returns the relevant list after having queried the Service Registry. The Planning Service then sends back the combined list of categories and SWS to the non-semantic service provider.

According to the list, a service integration request, including the name of the SWS and a list of the ontology entries is sent to the Planning Service, which forwards it to the Proxy Service to check if a SWS that is assigned all categories that describe the non-semantic web service can be found.

If yes, the non-semantic service is registered as the data source of the relevant SWS and the appropriate Service Registry entry is updated. If not, the SWS which shares most categories with the non-semantic web service is assigned the remaining categories. The Service Registry is updated, and the non-semantic service is registered as the data source of the relevant SWS. If no SWS match a single category in the Integration Request, a new SWS is created by the Proxy Service and the non-semantic service is registered as the data source of the relevant SWS

VIII. Example – Process Choreography

This chapter presents an overview of the *business process* involved in carrying out the use-case scenario described in chapter IV (Fig. 7). The process starts with an end-user's request to obtain information. If we recall the example scenario, amongst the information requested were upcoming events, event statistics, real-time traffic conditions, the availability of public transportation, an optimized travel-route between certain events, and the closest emergency facilities.

As the WikiCity system receives input from the end-user, it queries a number of services for different kinds of data, merges the data, and provides the resulting information back to the end-user.

The left side of Fig 7 illustrates a diversity of input-devices that end-users can interact with in order to access the WikiCity system. On the right side of Fig. 7, the different services involved in the use-case scenario are shown. These are either Semantic Web Services or simple databases that contain end-user-generated data. In both cases the data in question must be described in the WikiCity ontology for it to be accessible through the system, as explained in VII.D.

In order to fuse different types of data that are available in WikiCity, User Services are deployed to select relevant data types and identify the appropriate services that should be used to process the data. This results in a data-processing choreography, as described in VII.B. Finally, the entire business process can be executed by the User Service, as shown in VII.C.

A first implementation of the WikiCity concept was presented during the White Night event (la Notte Bianca) held in the City of Rome, Italy on September 8, 2007 (see http://senseable.mit.edu/wikicity/rome). This demonstration included a visualization of several real-time data streams that

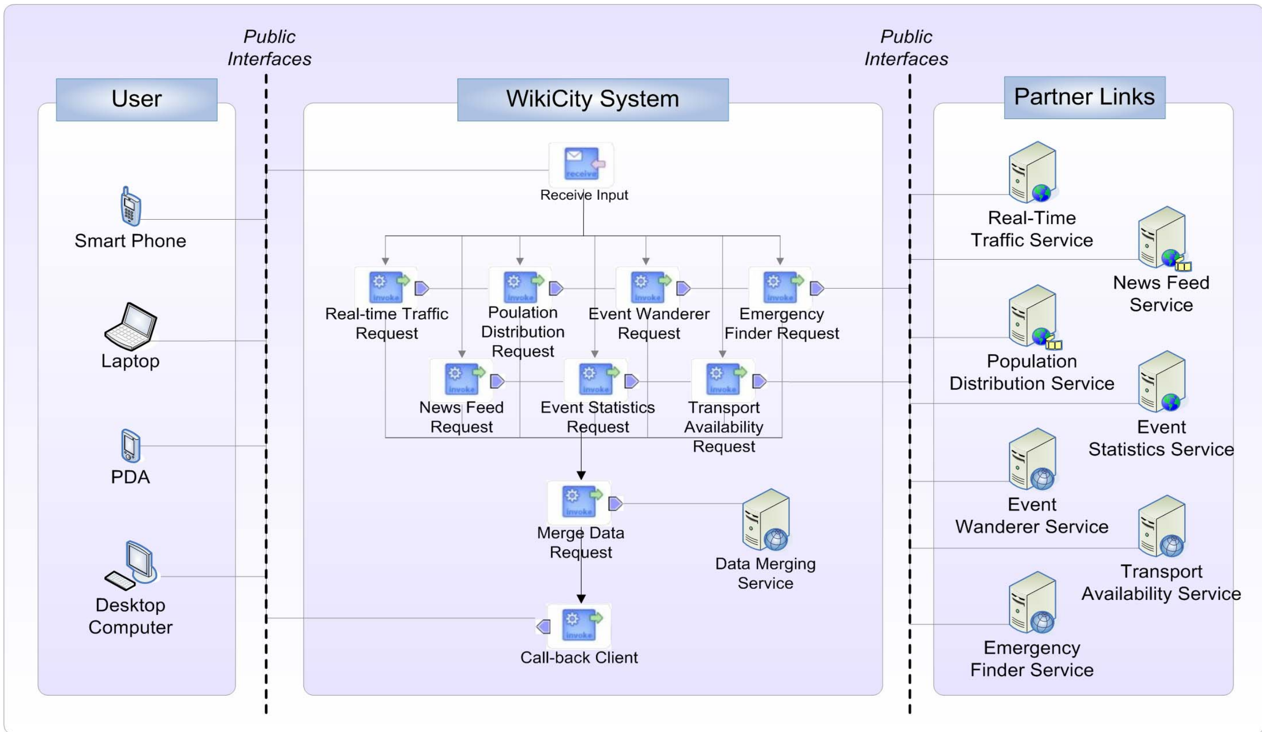

Fig. 7. Business Process for the Event Series Use-Case.

described major aspects of the event: distribution of population in real-time generated by analyzing cell phone activity, the position of buses and trains throughout the City, a live news feed from a major Italian newspaper, and real-time information about events that took in the City provided by a large yellow pages company.

IX. Conclusion

This paper presents the WikiCity concept, which aims to create a digital representation of a real-world city. Deployment of such a system will allow for more efficient use of city resources, enhanced social interaction, and improved emergency response. It could also enhance the perception of the city as an infrastructure for participation, social inclusion, and collaboration.

The WikiCity infrastructure includes three main components: data; a data-fusion engine which includes workflow management-services and user-tailored processes for combining heterogeneous data sources, and a back-end system that incorporates a semantic-web ontology for linking different data objects and services.

Two main request procedures can be performed by the end-user of the system: a data query can be submitted through the Semantic Web Services (SWS), and data can be combined by creating a business choreography using Processing Services.

An important benefit of the WikiCity system is its portable-base framework. It employs a modular structure with clear functional organization which allows easy application of the system to new use-case scenarios. The WikiCity system can be expanded through a simple process of introducing new Semantic Web Services and Processing Services.

Such flexibility is also one of the big challenges of the system. It requires a back-end with a service infrastructure and a comprehensive ontology, which is difficult to construct. In order to simplify the task, the ontology will be built at first for a narrow knowledge domain with particular use-cases in mind.

References

[1] Jonathan Reades, Francesco Calabrese, Andres Sevtsuk, Carlo Ratti, "Cellular Census: Explorations in Urban Data Collection," *IEEE Pervasive Computing*, vol. 6, no. 3, pp. 30-38, Jul-Sept, 2007

[2] Jennifer Thom-Santelli, "Mobile Social Software: Facilitating Serendipity or Encouraging Homogeneity?," *IEEE Pervasive Computing*, vol. 6, no. 3, pp. 46-51, Jul-Sept, 2007.

[3] Arianna Bassoli, Johanna Brewer, Karen Martin, Paul Dourish, Scott Mainwaring, "Underground Aesthetics: Rethinking Urban Computing," *IEEE Pervasive Computing*, vol. 6, no. 3, pp. 39-45, Jul-Sept, 2007.

[4] Jonathan Trevor, David M. Hilbert, "AnySpot: Pervasive Document Access and Sharing," *IEEE Pervasive Computing*, vol. 6, no. 3, pp. 76-84, Jul-Sept, 2007.

[5] Masatoshi Arikawa, Shin'ichi Konomi, Keisuke Ohnishi, "Navitime: Supporting Pedestrian Navigation in the Real World," *IEEE Pervasive Computing*, vol. 6, no. 3, pp. 21-29, Jul-Sept, 2007.

[6] Tarek Abdelzaher, Yaw Anokwa, Péter Boda, Jeff Burke, Deborah Estrin, Leonidas Guibas, Aman Kansal, Samuel Madden, Jim Reich, "Mobiscopes for Human Spaces," *IEEE Pervasive Computing*, vol. 6, no. 2, pp. 20-29, Apr-Jun, 2007.

[7] Nicolar Guarino, “Formal Ontology in Information Systems”, IOS Press, 1998.

[8] Peter Mika, “Ontologies Are Us: A Unified Model of Social Networks and Semantics”, Proceedings of the 4th International Semantic Web Conference (ISWC 2005), LNCS 3729, Springer-Verlag, 2005.

[9] Yang Li, Jason I. Hong, James A. Landay, "Design Challenges and Principles for Wizard of Oz Testing of Location-Enhanced Applications," *IEEE Pervasive Computing*, vol. 6, no. 2, pp. 70-75, Apr-Jun, 2007.