TELKOMNIKA, Vol. 11, No. 5, May 2013, pp. 2780~ 2790
ISSN: 2302-4046

An SOA-based Noise Mapping Platform for Urban Traffics

Wensheng Xu*[1], Nan Li[2]
[1]School of Mechanical, Electronic and Control Engineering, Beijing Jiaotong University, Beijing, China
[2]School of Material and Mechanical Engineering, Beijing Technology and Business University, Beijing, China
*Corresponding author, e-mail: wshxu@bjtu.edu.cn[1], linan@th.btbu.edu.cn[2]

Abstract

Traffic noise is a major environmental problem in many urban areas and frequently causes complaints from urban residents. An accurate traffic noise map of urban areas can facilitate noise monitoring, traffic strategic planning, street planning, residential area planning and noise prevention or reduction. An SOA based platform for urban traffic strategic noise mapping is proposed in this paper. Service Oriented Computing Environment (SORCER) is adopted to build the highly flexible distributed platform for noise monitoring and noise mapping. The platform architecture and the hierarchical services structure based on SOA are presented. The major services in the platform, including the task scheduler service, prediction service and noise propagation calculation service are analyzed in details. To demonstrate the function and mechanism of the platform, a real traffic noise mapping project for a Beijing area is presented.

Keywords*: computer aided noise mapping, traffic noise prediction, service oriented architecture, distributed computing*

1. Introduction

Traffic noise is a major source of noise pollution in many urban areas and often causes environmental complaints in residential areas. Noise disturbance significantly impacts many areas with high population density and affects the residents adversely in their daily life. More and more complaints and requests to reduce noise in big cities indicate that noise problems are very serious in these metropolitan areas. In the constituting process of noise control policies, traffic strategic noise maps are needed to survey existing distributions of noise levels and identify primary noise sources [1] or to predict noise levels for specific areas when planning public traffic routes. A computer aided noise mapping system is extremely necessary for these tasks.

Nowadays, the evolution of computer aided tools has made the road traffic noise mapping more precise and convenient. There currently exist a number of commercial tools such as CadnaA [2] and SoundPlan [3] which may be used to develop strategic noise maps in an effort to satisfy parts of the requirements from different kinds of users. A noise mapping system should generate traffic strategic noise maps for different purposes [4]: (1) Quantify main sources of noise; (2) Clearly illustrate environmental noise exposure to provide a reference for noise-control policy makers; (3) Facilitate the development of policies for controlling noise and enforcing the control of noise; (4) Draft a cost-benefit plan to assist districts desiring to reduce noise levels; (5) Adopt theories to examine the effects of environmental improvement plans; (6) Improve the enforcement of regional or national plans to decrease new noise sources as well as to protect new noise-sensitive and tranquility-needed areas; (7) Monitor noise reduction schemes and their effectiveness during the enforcement process; (8) Monitor changing trends in environmental noise; (9) Provide a research platform for studying the effects of noise on the human body.

Due to the purposes mentioned above, different kinds of software, such as GIS (Geographical Information Systems), statistics software or data visualization package, are usually used to work along with the noise mapping system together. Therefore, it is a big

Received January 7, 2013; Revised March 20, 2013; Accepted March 28, 2013

challenge to integrate all useful software tools as a loosely-coupled system that are extensible, flexible and fit well with existing legacy systems.

In general terms, Service Oriented Architecture (SOA) [5], [6] is widely considered to be the best practice when solving distributed and integration calculation problems mentioned above. SOA is a software architecture using loosely coupled software services that integrates them into a distributed computing system by means of service-oriented programming. Different from the client-server architecture which separates a client from a server, SOA introduces a third component called service registry. In SOA, the client is referred to as a service requestor, the server is referred to as a service provider and the service registry is introduced as an intermediate component. The service requestor and the service provider can find the service registry by the discovery and join protocol, then the service provider locates one or more service registries to advertise its availability and publish services, and the service requestor looks up a service in the service registry by search criteria related to the service name/type and quality of service. After finding a service, the service requestor binds to the service provider to execute the service.

In this paper, an SOA based platform for urban traffic strategic noise mapping is presented. The key modules of the platform are implemented as services which are loosely coupled in order to provide flexible specific functions. The service wrapping mechanism and procedure of services are discussed, and a real traffic noise mapping project is introduced to demonstrate the platform.

2. SOA-Based Noise Mapping Platform

2.1. Service-Oriented Computing Environment

Two popular implementations of SOA are Web Services and JINI which have many applications [5]. Transparent services in the noise mapping platform can be achieved with the SOA architecture. One of the first SOA applications in engineering was developed under the sponsorship of the National Institute for Standards and Technology (NIST)—the Federated Intelligent Product Environment (FIPER) [7]. The Service-Oriented Computing Environment (SORCER) is built on top of FIPER to introduce a metacomputing operating system with all basic services necessary, including a federated file system, to support service-oriented programming [5]. It is based on the JINI technology and provides an integrated solution for complex metacomputing systems.

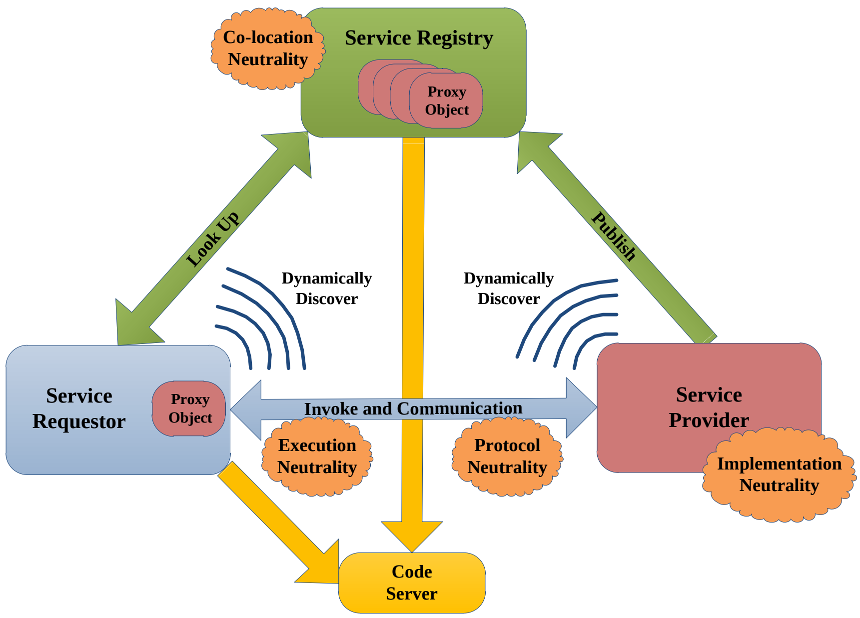


Figure 1. Working mechanism of services in JINI/SORCER

Figure 1 illustrates the basic features of JINI/SORCER. JINI/SORCER can differentiate from other SOA technologies (such as Web Services) according to whether or not the communication protocol between service provider and service requestor can be varied in different services. In JINI/SORCER, the communication protocol is not a fixed one, and both sides can choose the most efficient protocol(s) according to their needs, even binary protocol. Not only the location of the service provider can be dynamic, but also the location of the service registry can be dynamic as well. The features of JINI/SORCER can be discussed as four neutralities being compared with Web Services:

(1) Location neutrality—the locations of the service providers and service registries do not need to be fixed and static and do not need to be known to service requestors in advance;

(2) Execution neutrality—the execution of services in the service providers does not need to be known to service requestors and can be invoked by service requestors with ease of use;

(3) Protocol neutrality—the communication protocol between the service requestor and the service provider does not need to be known to the service requestor and it is determined solely by the service provider and is implemented by the SOA service proxy mechanism;

(4) Implementation neutrality—the service provider can be implemented with Java in JINI/SORCER and can run on any operating systems including Windows, Linux and MacOS, and the service requestors do not need to know the implementation details of the service provider.

These excellent features of JINI/SORCER provide good basis for the development of the noise mapping system which is comprised of a number of distributed components.

2.2. Platform Architecture

Based on JINI/SORCER technology, an SOA-based loosely-coupled platform for urban traffic strategic noise mapping is proposed, as shown in Figure 2.

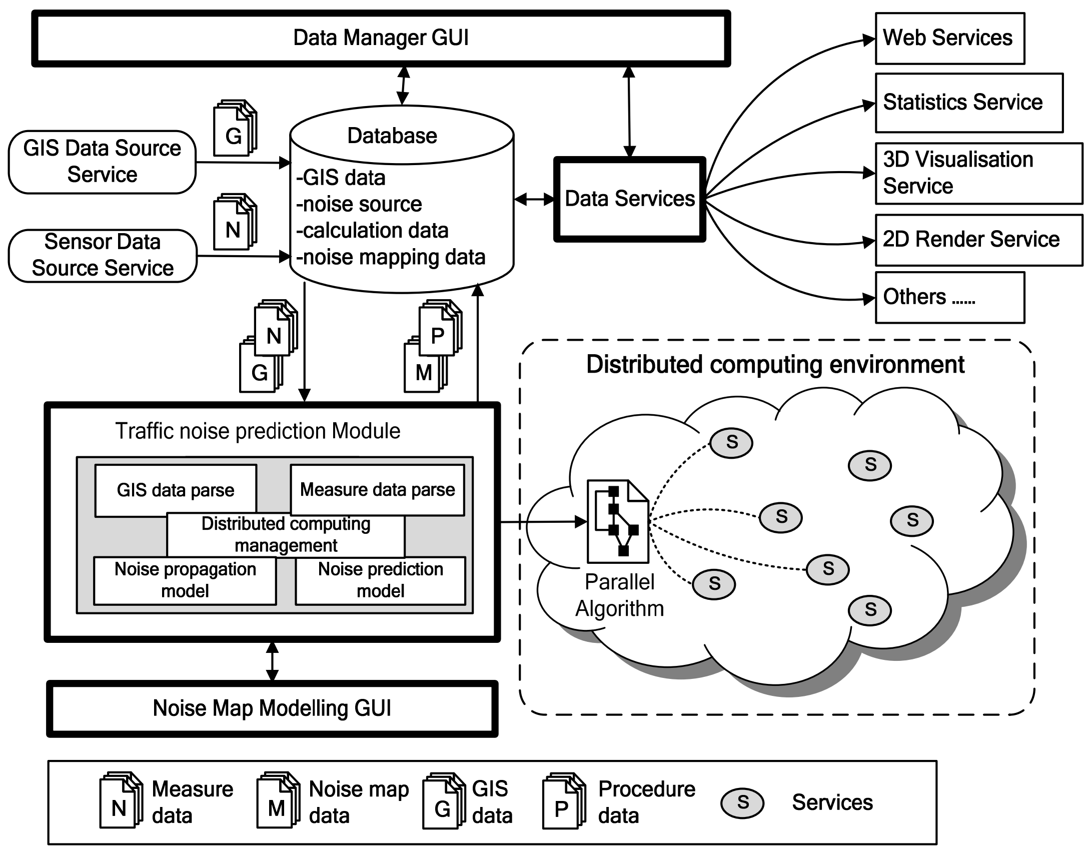


Figure 2. Platform architecture for urban traffic strategic noise mapping

The technology details of service registry, service lookup and service invocation will be hidden by JINI/SORCER. The platform only needs to deal with noise mapping service building, service management and mapping process control.

The design targets of the platform include:

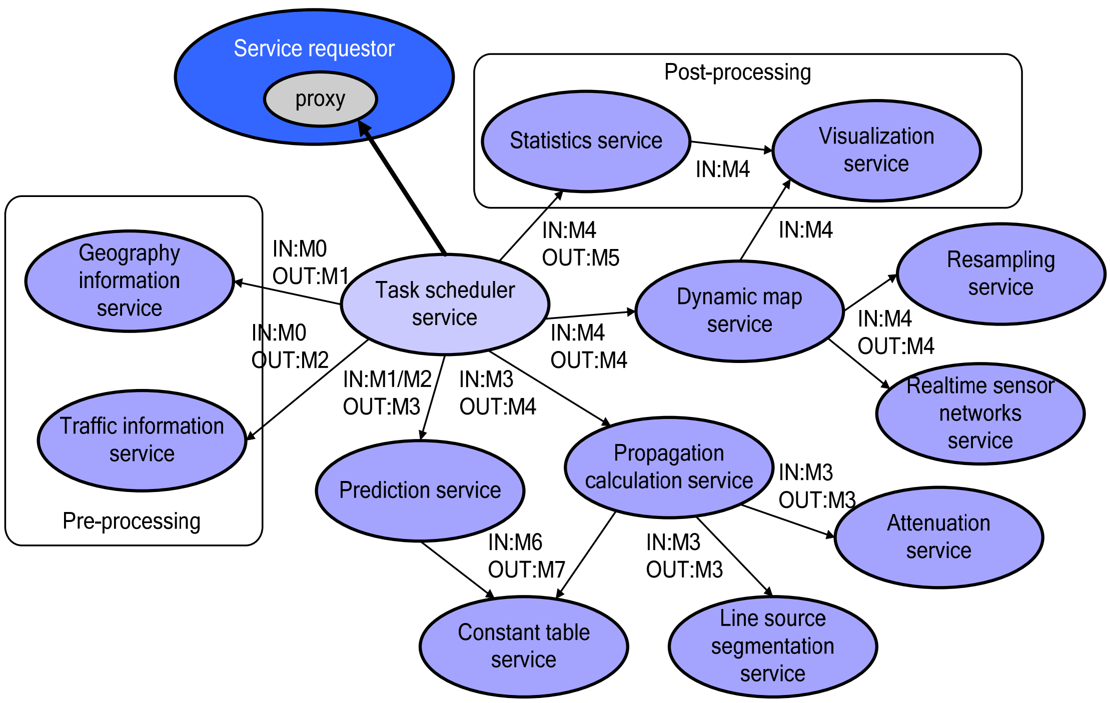
- To implement a sound objects modeling editor. In the platform, some traffic noise propagation associated data objects will be defined, including noise receivers, point sources, roads, buildings, barriers, line sources, calculation areas, auxiliaries, ground contours, ground absorption areas etc. This is the basis for urban area sound field modeling on top of the geographic map of a specific area. With the sound objects modeled on top of the geographic map, the noise effects can be calculated precisely with the noise propagation calculation model mentioned below.
- To import, export and convert general data. Different kinds of data, including GIS data, measure data and noise mapping result data, need to be recognized and processed in the platform. The urban traffic information must be converted into sound objects and employed by the noise prediction module. This is a basic functional module of the noise mapping platform.
- A flexible traffic noise prediction algorithm library. The prediction algorithm library can supply a general template to match different prediction models. A number of traffic noise prediction algorithms can be developed and stored in the noise prediction algorithm library and can be adopted and invoked by the platform, and can be compared and analyzed for their effects to accommodate different requirements.
- Noise propagation calculation model. In the platform, all of the noise propagation attenuation terms will be calculated from the method outlined in ISO 9613-2 [8], which include geometrical divergence, atmospheric absorption, diffraction, ground effect, etc. This noise propagation calculation model is implemented in the platform.
- Grid calculation and visualization for noise mapping. In order for end users to visually see the effects of the noise environment of a certain area, the noise effects of the certain area should be calculated and presented in graphics intuitively. Since a city area is normally so large and the scale of the calculation is so large, so a large area should be divided into a group of small grid areas and each grid area is carrying out the calculation and visualization and the final results are integrated together accordingly. So the grid calculation for traffic noise attenuation is the foundation of the whole noise mapping visualization. With the grid calculation pattern, a complex and large-scale calculation problem is decomposed into a group of simpler and smaller calculation problems and can be solved in parallel individually by a number of distributed computers.

To achieve the above targets, a platform GUI, a data manager GUI, a traffic noise prediction module, data services, a distributed computing environment and an underlying database system are designed and developed for the platform. All relevant data are stored in the database system, including GIS data, noise source data, calculation data and noise mapping data. The GIS data source service is responsible for collecting and importing GIS data, and the sensor data source service is responsible for monitoring, collecting and importing the measured noise data from streets. The noise map modelling GUI can provide a visualized editor for sound object modeling and the traffic noise prediction module. The traffic noise prediction module provides the noise propagation model and noise prediction model for the system and also manages the distributed computing environment for propagation and prediction calculation in the platform. The noise propagation calculation and prediction calculation are based on large amount of data and need complex data processing procedures, and a complex calculation task can be divided into smaller tasks and then be carried out by the distributed computing environment in parallel, and the results of the smaller tasks can be aggregated into the final result which can also be imported into the database system for subsequent processing. A parallel algorithm is adopted in the computation in the distributed computing environment. Data processing functions in the platform, such as statistics, 3D visualization and 2D rendering, etc. are all implemented as individual services and can be invoked by the platform through the data services module. Users of the platform can manipulate and manage the data in the platform through the data manager GUI.

2.3. Hierarchical Services Structure for Noise Mapping

Figure 3 illustrates the hierarchical services structure of noise mapping, which include 13 services: task scheduler service, geography information service, traffic information service, prediction service, constant table service, propagation calculation service, line source segmentation service, attenuation service, dynamic map service, realtime sensor networks service, resampling service, statistics service, visualization service. Some services provide specific functions for other services, and all services are integrated in the platform and are connected through certain invocations among them.

The geography information service and traffic information service act as the pre-processing stage for date processing in the platform, and the statistics service and visualization service act as the post-processing stage for data processing in the platform. The task scheduler service, prediction service and propagation calculation service are the core services in the platform and will be discussed in details below.



M0 – Target area
M1 – Geography information
M2 – Traffic information
M3 – Sound objects
M4 – Grid data of noise map
M5 – Statistics report
M6 – Parameter names
M7 – Parameter values

Figure 3. Hierarchical services structure for noise mapping

There are two user roles that users can play in the platform: service provider and service requestor. As a service provider, a user should supply the target area with the geography information and traffic information (the geography information service and the traffic information service shown in Figure 3), so the geographic map and noise sources can be modeled. To be a service requestor, a user can initiate and monitor the whole calculation process and get the final noise mapping result. Every service is autonomous and can invoke other services in the platform. The employer service does not need to care about what happen in the employee services, even though the employee services may call other services iteratively as well.

As shown in Figure 3, the task scheduler service is an integrated service to supply the whole function of noise mapping. The user only needs to invoke this single service to start the calculation process. Every invoking between services should follow matchable input/output message interfaces (message M0 to message M7 in Figure 3). In this way, all services can be individually developed, deployed and managed, and they are loosely coupled and can be invoked through the message interfaces, thus flexible architecture and easy management of the noise mapping platform can be achieved.

3. Services for Noise Mapping

3.1. Task Scheduler Service

Just like a hinge, the task scheduler service plays an important role in the platform. In the pre-processing stage (shown in Figure 3), the geography and traffic information will be imported and the noise mapping task will be created in the task scheduler service. In post-processing, the task scheduler service supplies result data to the statistics service so that the policy makers can get meaningful information from the noise mapping results.

To handle a large size traffic noise mapping calculation task, the total project is first subdivided into subtasks with rectangle tiles that fit together and then saved to database. Then the task scheduler service plays as a service requestor to invoke the service proxy and send the subtask information to the noise propagation calculation service provider. After that, the service provider loads independently one part of the project limited by one of the rectangle tiles, and the subtasks can be calculated in parallel with several service providers. The task scheduler automatically organizes and manages the required service proxy.

Multiple propagation calculation service providers can exist in the platform and can dynamically join and leave the platform under the mechanism of SORCER, and they can be assigned with the calculation subtasks. A delicate scheduling algorithm can be adopted inside the task scheduler to manage the calculation service providers considering the busy or idle status of the calculation service providers and their calculation efficiency.

3.2. Prediction Service

Nowadays, different prediction models are employed to build the urban traffic noise map, which include FHWA (Federal Highway Administration) model [9] in United States, CoRTN (Calculation of Road Traffic Noise) model [10] in Britain, ASJ (Acoustical Society of Japan) RTN-Model [11] in Japan and some other model etc [12]. To accommodate all these prediction models, a flexible adapter is needed in the noise mapping platform to work with all these prediction models potentially.

The prediction service supplies a parameter adapter to match different prediction models, as shown in Figure 4. The match mechanism for prediction models implies that the service space may include more than one prediction service, the services implement different prediction models but supply the same parameter input or output interface.

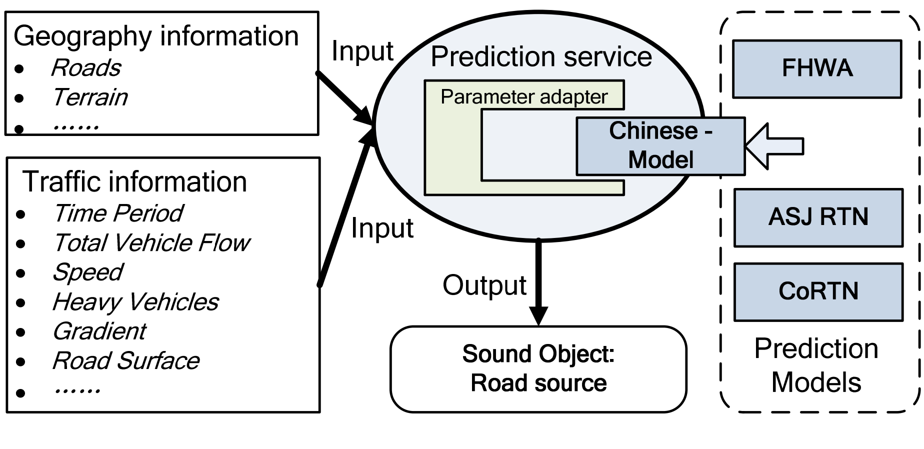


Figure 4. Prediction service

The input information includes geography information and traffic information (shown in Figure 4), and the output of the prediction service is the road source which is implemented as a sound object. The service requestor does not need to know the implementation details of the prediction model in the service invoking. And they even do not need to know which model has been employed in the chosen service. The prediction model can be chosen by users of the noise mapping platform through the platform GUI. Geography information such as roads information and terrain information, and traffic information such as time period, total vehicle flow, speed of vehicles, number of heavy vehicles, gradient, road surface circumstances, etc. will be provided for the prediction service.

3.3. Noise Propagation Calculation Service

In the noise propagation calculation service, the noise propagation attenuation terms include geometrical divergence, atmospheric absorption, diffraction, ground effect and so on, as shown in equation (1) and (2).

$$L_{fT}\left(DW\right) = L_W + D_C - A \tag{1}$$

L_W is the octave-band sound power level, in decibels, produced by the point sound source relative to a reference sound power of one picowatt (1 pW);

D_C is the directivity correction, in decibels, that describes the extent by which the equivalent continuous sound pressure level from the point sound source deviates in a specified direction from the level of an omnidirectional point sound source producing sound power level L_W. D_C equals the directivity index D_I of the point sound source plus an index D_Ω that accounts for sound propagation into solid angles less than 4π steradians; for an omnidirectional point sound source radiating into free space, $D_C = 0\ dB$;

A is the octave-band attenuation, in decibels, that occurs during propagation from the point sound source to the receiver.

The attenuation term A in equation (1) is given by equation (2):

$$A = A_{div} + A_{atm} + A_{gr} + A_{bar} + A_{misc} \tag{2}$$

A_{div} is the attenuation due to geometrical divergence;

A_{atm} is the attenuation due to atmospheric absorption;

A_{gr} is the attenuation due to the ground effect;

A_{bar} is the attenuation due to a barrier;

A_{misc} is the attenuation due to miscellaneous other effects.

Figure 5 illustrates the flow chart of the receiver calculation procedure. The input includes useful sound objects which contain all information needed in calculation, and the output is the result of the receiver calculation.

The noise calculation service depends on the line source segmentation service, attenuation service and constant table service, as shown in Figure 3. Sometimes, the calculation of the attenuation terms has to employ the geometry topology algorithm, for instance, the reflections effect calculation and the diffraction attenuation caused by barriers depend on the geometry calculation of the reflection sound beam paths and the direct sound beam paths. Due to that, the noise propagation calculation package will include two parts: acoustics model and geometry algorithm, as shown in Figure 6.

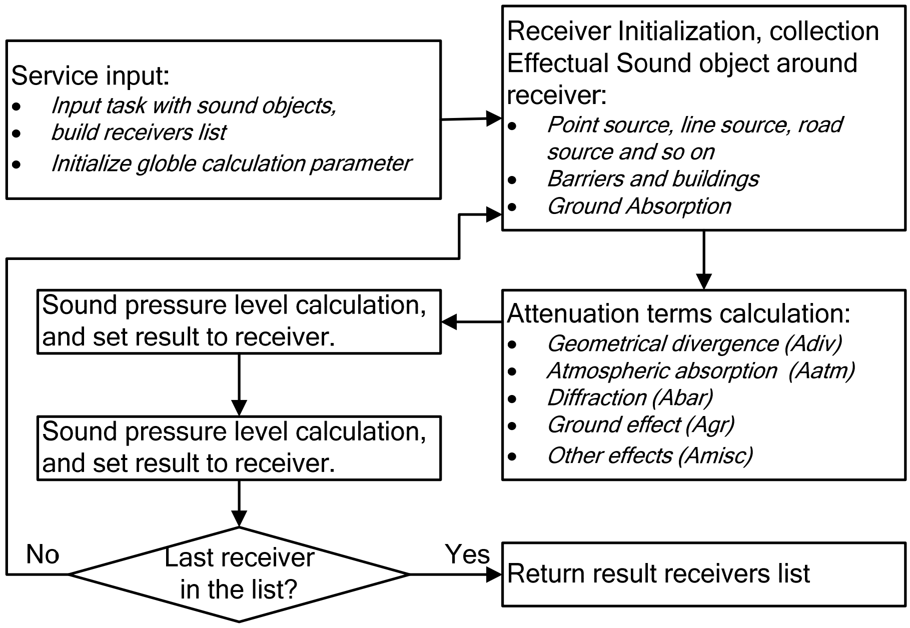


Figure 5. Procedure of receiver calculation

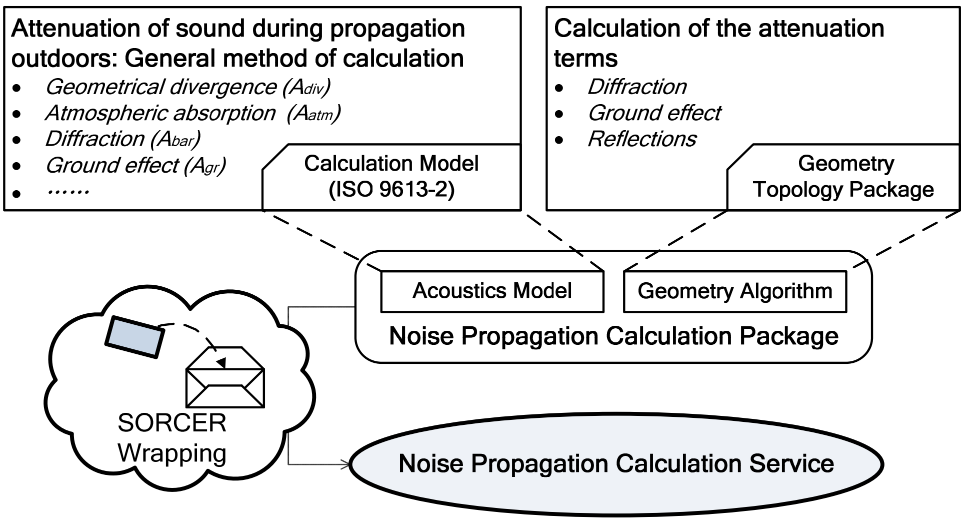


Figure 6. Noise propagation calculation service wrapping

In order for the propagation calculation capabilities to be provided as services to other requestors, the noise propagation calculation package will be wrapped as a SORCER service to become network objects which are always ready to be called by service requestors, as shown in Figure 6. This is called the SORCER wrapping step. The technology details of the service registry, service lookup and service invoking will be hidden and taken care of by SORCER. The user only needs to deal with services building and services management. With the SORCER service interface, the calculation services can work as service providers and join the platform dynamically and be invoked by the task scheduler service for noise propagation calculation tasks.

4. Platform Application

All services shown in Figure 3 have been implemented in the platform for demonstration. Figure 7 illustrates the demonstration of the noise propagation calculation service, the grid calculation result of a line source of noise propagating over 2 barriers. With the visualized graphics interface showing the calculation result, the darker color along the road implies higher level of noise. And accurate noise levels of each location in the map can be measured and got through the user interface. Figure 8 shows the demonstration of the prediction service and noise propagation calculation service. In this task, real geography information and traffic information of the demonstration area are imported to drive the noise propagation calculation. With the graphical user interface, the predicted noise level of a specific location in the map can be measured and showed.

In Figure 9, a real traffic noise mapping application is shown. In this example, a demonstration area (12 square kilometers in Beijing city) noise map is generated using the platform. The calculation configurations of the demonstration task are: (1) Area: 12 square kilometers; (2) Calculation grid size: 20 m; (3) Search radius: 1200m; (4) Calculation grid height: 54m; (5) Amount of buildings: 15674; (6) Amount of roads: 314; (7) Amount of subtasks divided: 4; (8) Calculation time: approximately 6.5 hours. The noise distribution in the area is calculated through noise propagation calculation and intuitively shown in the map. From this real noise map, the effects of road traffic planning and selections of residential area can be well shown and understood, and also noise prevention measures such as constructing noise barriers can be simulated through the noise mapping platform to verify their effects. In the user interface of the platform, the zoom in and zoom out functions are also provided show the details of the noise map in different detail levels.

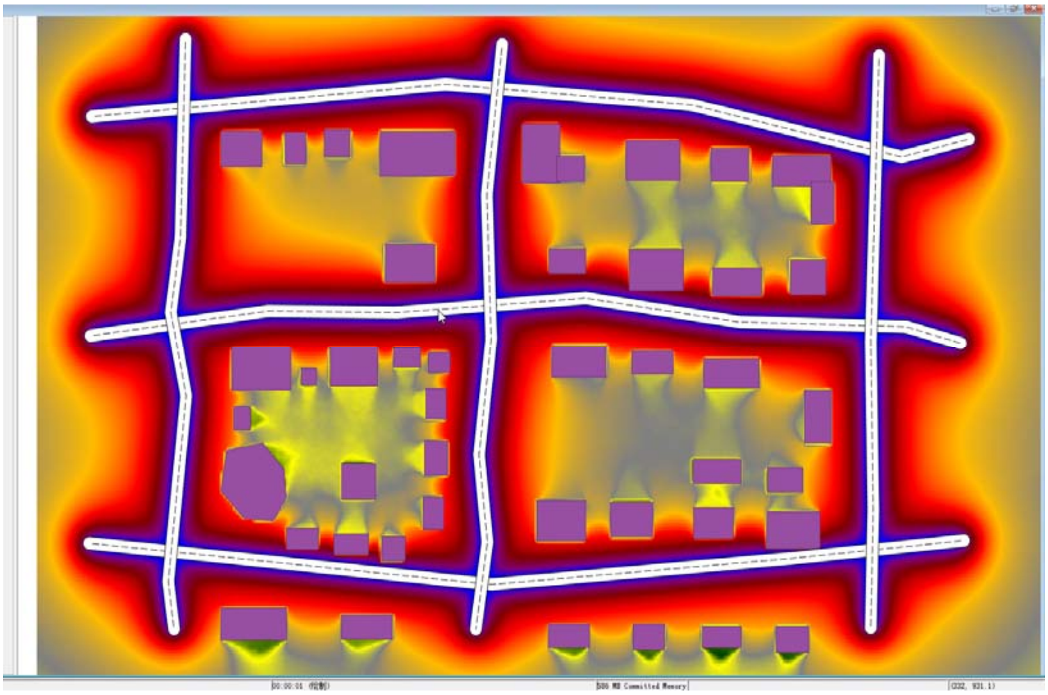


Figure 7. Demonstration of noise propagation calculation service (line source with barriers)

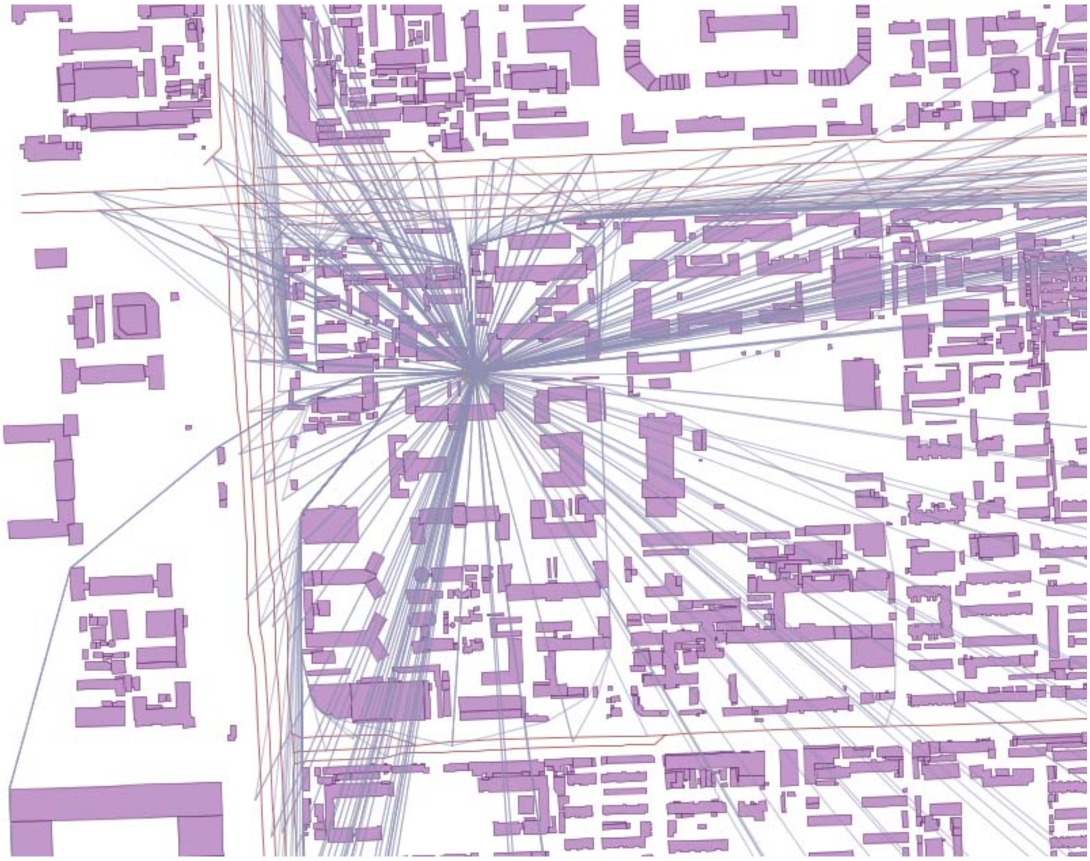


Figure 8. Demonstration of prediction service and noise propagation calculation service (road source)

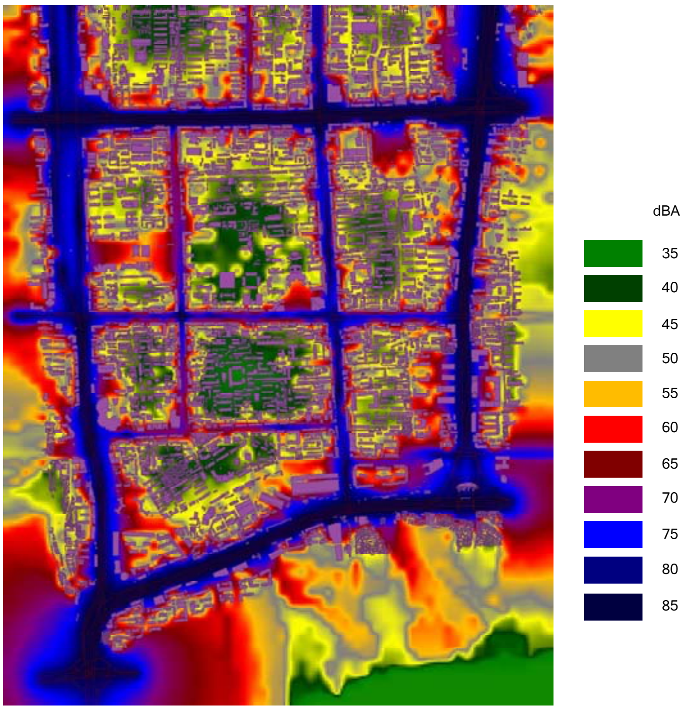


Figure 9. Traffic noise map of a demonstration area in Beijing

5. Conclusion

The work described in this paper presents details of an SOA-based loosely-coupled platform for urban traffic mapping. This platform can supply a flexible mechanism to integrate different useful services in noise mapping calculation. The whole solving process, consisting of Task Scheduler Service, Prediction Service and Noise Propagation Calculation Service, can import and manage traffic data and generate good-quality calculation solutions in a reasonable period of time. The technology details of the service registry, service lookup and service invoking are hidden by SORCER technology.

A real traffic noise mapping project is introduced to demonstrate the services and the platform mentioned above. From the time (6.5 hours) needed for calculating the noise mapping of this 12 square kilometers area we can know that the computation facilities and computation efficiency are not efficient enough for traffic noise mapping in very large-sized areas which will cause tremendous cost of calculation time. Therefore apart from adopting more powerful computer servers for the noise mapping platform, the future work will focus on efficiency improvement for the calculation services in the platform.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (51175033), and Funding Project for Human Resources Development in Institutions of Higher Education of Beijing, China (PHR201107110) and general program of Science and Technology Development Project of Beijing Municipal Education Commission (KM201210011007).

References

[1] Tsai KT, Lin MD, Chen YH. Noise mapping in urban environments: A Taiwan study. *Appl Acoust*. 2009; 70(7): 964-972.

[2] Yang W, Kang J. Soundscape and Sound Preferences in Urban Squares: A Case Study in Sheffield. *Journal of Urban Design*. 2005; 10(1): 61-80.

[3] Rossie F, Umberto Di M, Sofia S. *An original method for urban traffic noise prediction*. Proceedings of Euro-noise 2006. Tampere, Finland. 2006: 1-6.

[4] Department for Environment, Food and Rural Affairs (DEFRA, U.K.). Towards a national ambient noise strategy -- A Consultation Paper from the Air and Environmental Quality Division; 2001.

[5] Sobolewski M. *SORCER: Computing and Metacomputing Intergrid*. Proceedings of 10th International Conference on Enterprise Information Systems (ICEIS 2008). Barcelona, Spain. 2008: 74-85.

[6] Hermawan, Sarn R. Developing distributed system with service resource oriented architecture. *Telkomnika*. 2012; 10(2): 389-399.

[7] Sobolewski M. *Federated P2P services in CE Environments*. Proceedings of the 9th ISPE International Conference on Concurrent Engineering. Cranfield, U.K. 2002: 13-22.

[8] ISO. 9613-2:1996. *Acoustics – Attenuation of Sound during Propagation Outdoors–Part 2: General Method of Calculation*. Geneva, Switzerland: ISO Press; 1996.

[9] Anderson G, Lee C, Fleming G, etc. *FHWA Traffic Noise Model User's Guide*. Department of Transportation. Report number: FHWA-PD-96-009, DOT-VNTSC-FHWA-98-1. 1998.

[10] Department of Environment and Welsh Office UK (DoT, U.K.). Calculation of road traffic noise. London: HMSO. 1988.

[11] Yamamoto K. *Road Traffic Noise Prediction Model 'ASJ RTN-Model 2008': Report of the Research Committee on Road Traffic Noise*. *Acoust. Sci. & Tech*. 2010; 31(1): 2-53.

[12] Yu Y, Zhou L. Acoustic Emission Signal Classification Based on Support Vector Machine. *Telkomnika*. 2012; 10(5): 1027-1032.