

# The Implementation of OPENLS Presentation Service in Location Based System.

Tae-wook Heo, Jun-wook Lee, Sung-soo Kim, Jong-Hyun Park  
LBS team, ETRI

161 Gajeong-dong Yuseong-gu, Daejeon 305-350, Korea  
[htw398@etri.re.kr](mailto:htw398@etri.re.kr), [lhw63822@etri.re.kr](mailto:lhw63822@etri.re.kr), [sungsoo@etri.re.kr](mailto:sungsoo@etri.re.kr), [jhp@etri.re.kr](mailto:jhp@etri.re.kr)

**Abstract:** Location-based services allow consumers to receive services based on their geographic location data. For example, businesses can provide information to find traffic facility, public restaurant, a hospital, and a station based on the consumer's location at a particular moment. Also, they wish to grasp a accurate location based service in mobile device or PDA. So, this paper shows that presentation service offers a base-map to be overlaid OpenLS (Open Location Services)'s directory service, route determination service, geocode and reverse-geocode service. The presentation service is the portrayal of a map and portrayal of feature as map overlays..

**Keywords:** LBS, Web Service, Presentation Service.

## 1. Introduction

The importance of location based services grows highly as the development of mobile communication. For user's convenience, several location based services such of the route determination, directory, geocoding service are proposed. For this tendency, the program was implemented in this paper to present a map in mobile device based Java web service technology without Korean mobile communication companies' various platform. These services consisted of java based web services are available in various platform. These structures of EJB(enterprise java bean) are used to implement them and we simulated it by PC. And as the form of correspondent to the standard of OpenLS specification, a map presentation program was designed and implemented. Also, the opened LBS platform was supported, used the SOAP and XML as the data transformation standard.

## 2. Web Services

Now, the fundamental standards promoted on W3C are XML(Extensible Markup Language), UDDI(Universal Description Discovery and Integration), WSDL(Web Service Description Language), SOAP(Simple Object Access Protocol). And Fig. 1 showed the basic structure of web services. The XML schema defined web service's basic data format, UDDI used web services publish, advertisement, and outer web service's findings. WSDL provides the detailed information necessary to invoke a particular service. [4].

Fig. 2 shows that the service requestor asks the service registry to find its desired service. Then, the service registry searches the registration data of provider and sends the service provider's connection URL or information of service to the service requestor. So, the service requestor combines to the service provider through the connection URL, and receives services.[2]

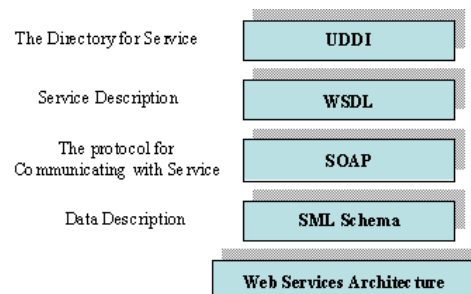


Fig. 1. The Structure of Web Service.

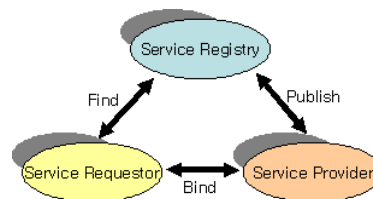


Fig. 2. Web Service Framework.

As previously stated, web service in this paper is based XML java web service in data implementation layer, therefore these technologies operates each other in the core layer, and supports various standard protocols(SMTP, HTTP, FTP). Also, they show input/output parameters, the structure of function, and service protocol binding using the WSDL stated web service's interface.

XML-RPC is a specification for representing remote procedure calls and results using XML in a simple yet powerful way. Fig. 3 explains the communication procedure of XML-RPC based Java language. If client calls a Java method, the method is serialized by XML, enveloped SOAP and transported using the standard protocols. And in response it can be accepted java method. In the procedure, it can be communicated either .NET or J2EE platform.[4],[5].

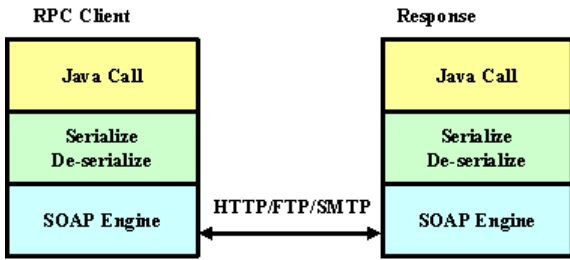


Fig. 3. The Procedure of XML-RPC

### 3. Presentation Service

#### 1) Presentation Service form

Presentations service provides a map using web service. Fig. 4 showed the whole system of presentation service. The system called a presentation service in the mobile device as a web service form connected a wireless network WIPI platform.

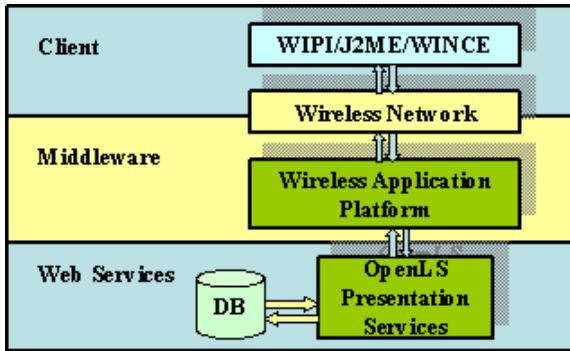


Fig. 4. Presentation Service based Architecture.

Fig. 5 shows a use-case diagram of presentation service. If the client requests the OpenLS's core services(route determination service, POI(Position Of Interest) service, and Position service), the system provide a map which overlaid these services to the base map using the web service for the users and implemented it through J2EE EJB.[2],[3]

And Fig. 6 shows a use-case diagram of presentation service's map generation. When it creates a map, the geo-informatics data is needed through DBMS. Oracle's spatial operation is useful for this service's implementation.

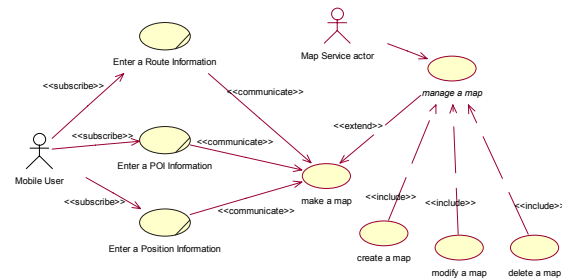


Fig. 5. Presentation Service Use-case Diagram.

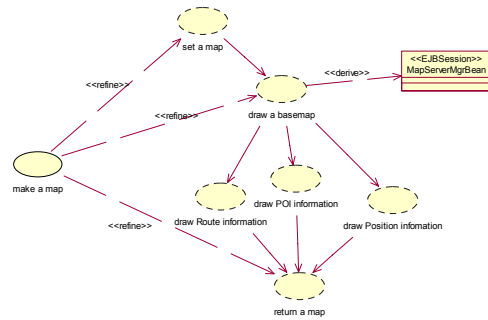


Fig. 6. Map Creation Diagram.

#### 2) Presentation Service Schema

Fig. 7 and Fig. 8 shows respectively request and response definition XML file of presentation service XML schema. The presentation service supported in this paper keeps OpenLS's proposed standard perfectly.[1]

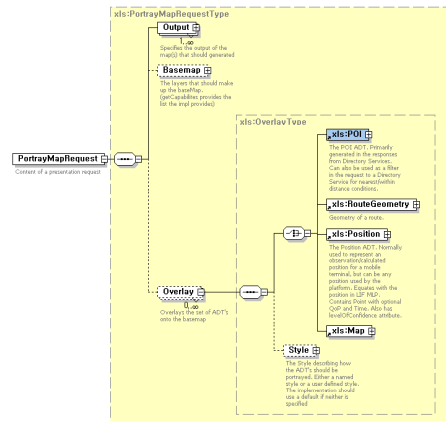


Fig. 7. Presentation Service Request Schema.

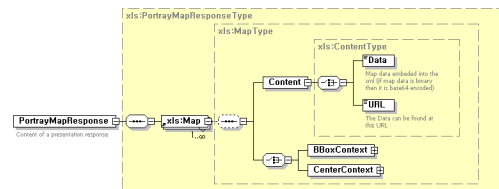


Fig. 8. Presentation Service Response XML Schema.

#### 3) Presentation Service Test Environment

JDK compiler version is Java SDK 1.3 version and web service toolkit is Apache AXIS toolkit. And DBMS is Oracle 9.2.0.1 supported spatial operation, WAS(Web Application Server) used IBM WebSphere Application Server 5.0, Java developer tool is WSAD(WebSphere Studio Application Developer).

#### Test Environment

- J2EE 1.3 EJB + J2SE 1.3
- SOAP : Apache AXIS
- DBMS : Oracle 9.2.0.1 + SDO
- WAS : IBM WebSphere Application Server 5.0

#### 4) Presentation Service

Fig. 9 shows the program flow chart of the presentation service. First, clients input the request of POI, Position, and Route information. And then input environment parameter and interest area, operate using DB spatial operation, and return the result of map data.

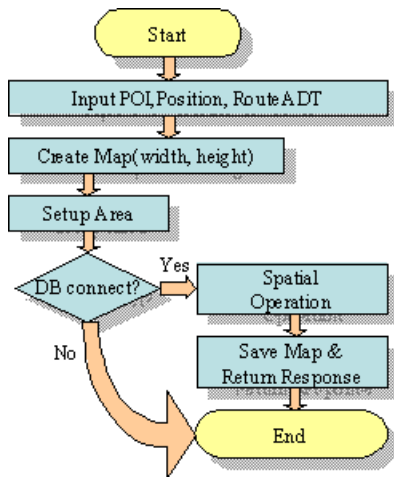


Fig. 9. Presentation Service Program Flow.

#### 4. Result & Discussion

Fig. 10 and Fig. 11 shows the OpenLS specification suggested XML files of PortrayMapRequest and PortrayMapResponse. Fig. 12 and Fig. 13 shows the result of the experiment. Compared with two files, they are likely to same as the specification.

```

PortrayMapRequest XML
<PortrayMapRequest>
  <Output width="640" height="480" format="image/png">
  <BBoxContext>
    <UpperCorner> <Coord cs="*"> -114.342 50.234 </Coord> </UpperCorner>
    <LowerCorner> <Coord cs="*"> -114.123 50.031 </Coord> </LowerCorner>
  </BBoxContext>
  <Output>
  <Basemap filter="Exclude" />
  <Overlay>
    <Position> -114.123 50.031 </Position>
  </Overlay>
</PortrayMapRequest>
  
```

Fig. 10. OpenLS PortrayMapRequest.

```

PortrayMapResponse XML
<PortrayMapResponse>
  <Map>
    <Content width="640" height="480" format="image/png">
    <data> ... </data>
  </Content>
  <BBoxContext>
    <UpperCorner> <Coord cs="*"> -114.342 50.234 </Coord> </UpperCorner>
    <LowerCorner> <Coord cs="*"> -114.123 50.031 </Coord> </LowerCorner>
  </BBoxContext>
  </Map>
</PortrayMapResponse>
  
```

Fig. 11. OpenLS PortrayMapResponse.

```

PortrayMapRequest XML
<PortrayMapRequest>
  <Overlay> <Position> <Point>
    <Coord cs="," decimal="," ts=""> -114.123 50.031 </Coord>
  </Point> </Overlay>
  <Output format="image/png" height="160" width="120">
  <BBoxContext>
    <UpperCorner> <Coord cs="," decimal="," ts=""> -114.342 50.234 </Coord> </UpperCorner>
    <LowerCorner> <Coord cs="," decimal="," ts=""> -114.342 50.234 </Coord> </LowerCorner>
  </BBoxContext>
  <Output>
  <Basemap filter="Exclude" />
</PortrayMapRequest>
  
```

Fig. 12. Web Service implemented Request XML

```

PortrayMapResponse XML
<?xml version="1.0" encoding="UTF-8" ?>
<PortrayMapResponse>
  <Map>
    <Content format="image/png" height="160" width="120">
    <Data>Data</Data>
  </Content>
  <BBoxContext>
    <UpperCorner> <Coord cs="," decimal="," ts=""> -114.342 50.234 </Coord> </UpperCorner>
    <LowerCorner> <Coord cs="," decimal="," ts=""> -114.342 50.234 </Coord> </LowerCorner>
  </BBoxContext>
  </Map>
</PortrayMapResponse>
  
```

Fig. 13. Web Service implemented Response XML

Now, it is going to be establishing the standards of OpenLS's specification. If the standard will be completed, original source codes reproduction are need for the modified standard.

#### 5. Conclusion

The implementation system in this paper shows the map presentation service based Java fitted for OpenLS specification and supported the architecture implementation of multi-platforms.

Yet, OpenLS specification is not standard but only request for comments. In the future, publish the completed the standard, we will implement the web service properly.

#### References

- [1] OpenLS Presentation Service, OpenGIS© Project Document (OGC 02-091), *Open GIS Consortium Inc.*, 11 November 2002.
- [2] E. Christenson, F. Curbera, G. Meridith, and S. Weerawarana, Web Services Description Language (WSDL) 1.1, *W3C Note (March 15, 2001)*, see <http://www.w3.org/TR/wsdl>.
- [3] Rod Johnson, J2EE Design and Development, *Wrox Press*, October, 2002.
- [4] David A. Chappell, Tyler Jewell, Java Web Services, *O'Reilly*, March, 2002.
- [5] H. Bequet, M.M. Kunnumpurath, S. Rhody, A. Tost, Beginning Java Web Services, *Wrox Press*, Sep, 2002.