

Online network analysis from heterogeneous datasets – Case study in London train network

Dr.Sanphet Chunithipaisan¹, Mr.Phil James², Prof.David Parker²

¹Department of Survey Engineering, Chulalongkorn University, Bangkok, Thailand.

²School of Civil Engineering and Geosciences, University of Newcastle upon Tyne, Newcastle upon Tyne, UK.

Email: Csanphet@chula.ac.th, Philip.James@ncl.ac.uk, David.Parker@ncl.ac.uk

Keywords: Interoperability, Network Connectivity, WebGIS, Standard and Specification, GML

Abstract

This paper reports research that resolves the issue of data integration from multiple heterogeneous datasets for performing network analysis operations. The current geospatial standards, protocols and technologies are investigated and implemented through the research. The methodologies to solve the creation of network topology on-line for supporting network analysis are suggested and tested. A software system is created with a number of tools to support such system. A scenario of application is tested around the real world dataset of the train network in London (UK).

1. Introduction

Utility and infrastructure networks impact society globally and are managed in both the private and public sectors. A huge amount of money has been invested to improve and develop these networks and their supporting systems to help service customers and solve business problems. The use of GIS for utility and infrastructure network is well-known and widespread. It provides a variety of tools to assist in the management and manipulation of utility and infrastructure networks. GIS can also support specialist tools for the analysis of linear networks. The specialised tools to manage and build topology are required to maintain the data and analyse network. The network GIS applications that use such tools include asset management, site selection, risk analysis and shortest path.

Sharing data among departments and/or organisations can save a huge amount of money. The most successful companies in the utility and infrastructure business use GIS to integrate geospatial with other corporate data to take maximum advantage of its resources (ESRI, 2002). There is, however, often little collaboration between organisations despite similarities of interest. This is mainly because of issues of data sharing and interoperability. These issues have concerned the geospatial spatial (GI) community for many years. Several organisations have presented a number of standards and specifications that attempt to persuade users to implement their system using the same standards or specifications. One such organisation is the Open GIS Consortium (OGC) which has developed and put forward a number of specifications and standards, aimed at promoting data sharing and dissemination amongst the GI community. Increasingly OGC specifications are being adopted and implemented by GI technology vendors and users to provide solutions to the problems of web based GI dissemination. In particular; the Geographic Markup Language (GML) (OGC, 2003) and web map services (e.g. Web Map Server (WMS) (OGC, 2001), Web Feature Server (WFS) (OGC, 2002)) looks set to play an increasingly crucial role in the future distribution of GI and services. The basic operation of requesting and retrieving data are performed via Uniform Resource Locators (URLs) and uses the Common Gateway Interface (CGI) protocol to pass the request details.

This paper reports the development of an expert system that is capable of combining linear network datasets from different data sources and carrying out network analysis taking advantage of existing standards, protocols and technologies. This system involves many various implementations and technologies. This research uses the

London train network (including rail, tube and tram) for a case study to test such a system.

2. The London train network

The London train network is one of the largest public city transportation networks in the world. There are about 1000km of track in the network and about 4 million journeys made every day (Transport, 2003a). It includes three modes of train-based system; rail, tram and tube (the name of the underground railway system). Table 1 shows the names of the network lines in each mode of transportation.

Mode	Name
Rail	Dockland Light Railway, National Rail
Tram	Tramlink
Tube	Bakerloo, Central, Circle, District, East London, Hammersmith & City, Jubilee, Metropolitan, Northern, Piccadilly, Victoria, Waterloo

Table 1. The network lines in London train network

The base route map provided by (Transport, 2003b) is in raster format which is a cartographic map showing the routes and connections of the train network in London. This base data was converted into vector format by digitising and then converted into GML format. Each train line (e.g. Bakerloo, Tramlink, Victoria) are stored in a separate GML file. Additional data of the station and interchange station are also created and stored in GML files.

3. The architecture of the software system

The aim of this research is to develop a software system that enables the analysis of network data that comes from multiple different data sources. To achieve this aim, we use the web to share data and utilise GML as the data format. The intelligent engine is required to retrieve data from different data sources through the web. Such an engine also needs to have the capability of getting the data request from the user about which datasets they want, and then deliver these data back to the user once all the required data has been retrieved. The software system also has to provide tools to build the network topology based on the network data depending on the rules of connectivity defined and to carry out the network analysis. The configuration document is required to allow the user to set-up the project configuration which includes the datasets required for an application and the rules of connectivity to build the network topology. The architecture of the software system is shown in figure 1.

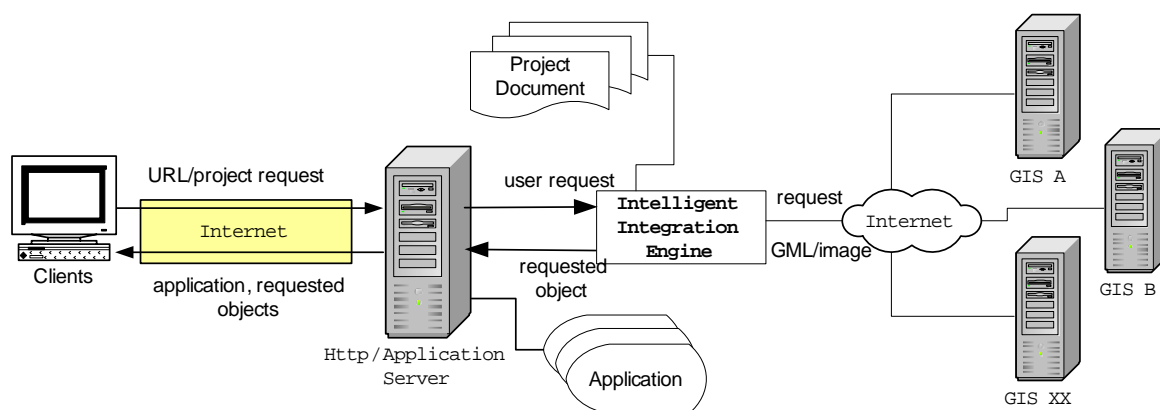


Figure 1. The software system architecture

As shown in figure 1, the user can open an online application and send a project request the intelligent integration engine. Such engine will send the request for each dataset to data server which is defined in the project document, retrieve these data and deliver them back to user. This engine also builds the rules of connectivity which are defined in the project document and sends it back to user. The network building and network analysis tool are resident in the application which allows the user to build network topology and carry out network analysis on-the-fly.

4. Research methodology

The software system designed involves many various protocols, technologies and developments that make it happen. Several parts that make up the system were implemented and developed which are discussed below.

4.1 Project document

The project document is the configuration document that notifies the system which datasets are required for an application and how features are connected. The project is documented using XML.

The information about a dataset used in an application includes the type of protocol, the name of feature, the type of data and the address of datasets. The following shows the sample of how project datasets are coded.

```
<rwo_dataset>
  <dataset protocol="http" name="rail" type="gml"
    address="http://site1.com/data/rail.gml"/>
  <dataset protocol="http" name="jubilee" type="gml"
    address="http://site2.com/data/wfs?obj=jubilee"/>
  <dataset protocol="http" name="circle" type="gml"
    address="http://site3.com/data/wfs?obj=circle"/>
  <dataset protocol="http" name="station" type="gml"
    address="http://site4.com/data/station.xml"/>
</rwo_dataset>
```

To define the rules of feature connectivity, we adopt the concept of the network family (Chunithipaisan et al., 2002) to describe how features are connected. A network family contains a set of various features types that make up a network. Feature types are defined in a group if they are the same kind of feature types e.g. road, street, avenue are grouped in road. As feature types are defined using a generic group name, the network family provides a tool to map feature types used in the databases to the generic group name used in the connectivity rules. Two documents are required for defining the network family. One is the rules of feature connectivity; another is the semantic of the feature types. The connectivity of features can be modelled as shown in figure 2.

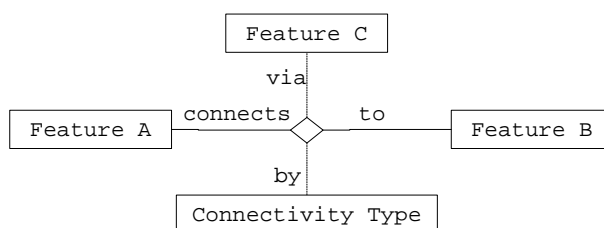


Figure 2. Feature connectivity model

As shown in figure 2, we can document the rule of feature connectivity in the network family in XML as follows.

```

<connrule>
  <rule from="rail" to="rail" via="station" type="40"/>
  <rule from="rail" to="tube" via="station" type="40"/>
  ...
</connrule>

```

As discussed previously, the names of features coded in the rules are the representative name of feature types which can be defined in the same common group. In order to link back to the database/dataset name used in an application, the document of semantic links is required. The sample of coding semantic links follows.

```

<semantic>
  <point_schema>
    <schema name="station">
      <object feature="station"/>
    </schema>
    ...
  </point_schema>
  <chain_schema>
    <schema name="tube">
      <object feature="bakerloo"/>
      <object feature="central"/>
      ...
    </schema>
    ...
  </chain_schema>
</semantic>

```

The relationship of documents which are required for each project can be shown in figure 3.

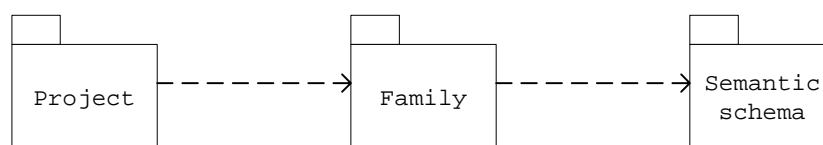


Figure 3. The software system architecture

As shown in figure 3, project document contains three separate XML documents that are interrelated. The project file contains a list of datasets required for the application. The family file supplies the rules of feature connectivity. The semantic schema provides the name mapping to the names of datasets used in the application. The link between project and family and family and semantic can be coded respectively as follows.

```

<family address="London_network.xml"/>
<schema address="semantic1.xml"/>

```

4.2 Intelligent integration engine

The intelligent engine to integrate data from multiple heterogeneous data sources via the web was developed using Java servlet technology. It allows users to send a request of which datasets they want via the web. When it receives a user request, it opens the project document and starts to open the web connection to the GIS data server defined in the project and return that data back to the system. When it has

retrieved all the data required then it combines those data together. It also converts the rules of feature connectivity defined in the project document into a form which is understandable to the application. After getting all data required, it delivers these objects back to the application.

4.3 Network topology builder

Network topology needs to be built to perform network analysis. It has to be built based on the rules of feature connectivity. It starts by cleaning geometries by intersecting and snapping the features and then builds the topology which contains a set of links and nodes for network family. With a set of links and nodes generated, this allows the application to perform network analysis. This tool was developed using Java technology.

4.4 Network analysis tool

The network analysis tool was developed which enables a user to find the shortest path between two points. The shortest path function was developed by implementing Dijkstra's (Dijkstra, 1959) algorithm. To determine the shortest path, the weight of the path is required. In general applications, the weight is usually the distance of the link between two nodes. However the weight can be any number attributes used to find the shortest path e.g. travelling time.

4.5 Application interface

An application was developed using Java applet technology. A Java applet is an application that can run on a web browser whilst the applet itself resides on a server. A number of Java classes were developed to support a customisation of the application system. Several tools of basic GIS rendering functions (e.g. zoom/pan) were developed.

5. The case study of London train network

The application of London train network was developed to find the shortest path for a journey between two stations. In this case, the weight for the shortest path is the distance of the geometry which is derived from the geometry of network features. An additional linear network feature called "alink" was set for a connection line between network features that can be interchanged at interchange stations. The rules of feature connectivity for London train network are defined as follows.

```
<connrule>
  <rule from="rail" to="rail" via="station" type="40"/>
  <rule from="rail" to="tram" via="station" type="40"/>
  <rule from="rail" to="tube" via="station" type="40"/>
  <rule from="rail" to="alink" via="station" type="40"/>
  <rule from="tram" to="tram" via="station" type="40"/>
  <rule from="tram" to="tube" via="station" type="40"/>
  <rule from="tram" to="alink" via="station" type="40"/>
  <rule from="tube" to="tube" via="station" type="40"/>
  <rule from="tube" to="alink" via="station" type="40"/>
  <rule from="alink" to="alink" via="station" type="40"/>
</connrule>
```

The semantics of network features are coded in the semantic schema as follows.

```
<semantic>
```

```
<point_schema>
  <schema name="station">
    <object feature="station"/>
    <object feature="station_interchange"/>
  </schema>
</point_schema>
<chain_schema >
  <schema name="rail">
    <object feature="railway"/>
    <object feature="dockland"/>
  </schema>
  <schema name="tram">
    <object feature="tramlink"/>
  </schema>
  <schema name="tube">
    <object feature="bakerloo"/>
    <object feature="central"/>
    <object feature="circle"/>
    <object feature="district"/>
    <object feature="east_london"/>
    <object feature="hammersmith"/>
    <object feature="jubilee"/>
    <object feature="metro"/>
    <object feature="northern"/>
    <object feature="piccadilly"/>
    <object feature="victoria"/>
    <object feature="waterloo"/>
  </schema>
  <schema name="alink">
    <object feature="station_link"/>
  </schema>
</chain_schema>
</semantic>
```

When the application starts up, it sends the request of the project of London train network to the integration engine servlet. This engine will read the project document. It then opens connections to data servers and retrieves all data defined in the project. It also creates a network family object which contains the rules of feature connectivity. Finally, all feature objects with the network family object are sent back to the application at the client side. Once the application gets all required objects, it will clean geometry and build the network topology. After building the network topology, it is ready to perform the shortest path analysis function.

The application shows the map of the London train network as a background. It provides a list of station for departure and destination stations and a shortest path function. The result of shortest path will be shown using a high-light graphic. After finding a shortest path, a report is generated to list a route and the stations in the shortest path result.

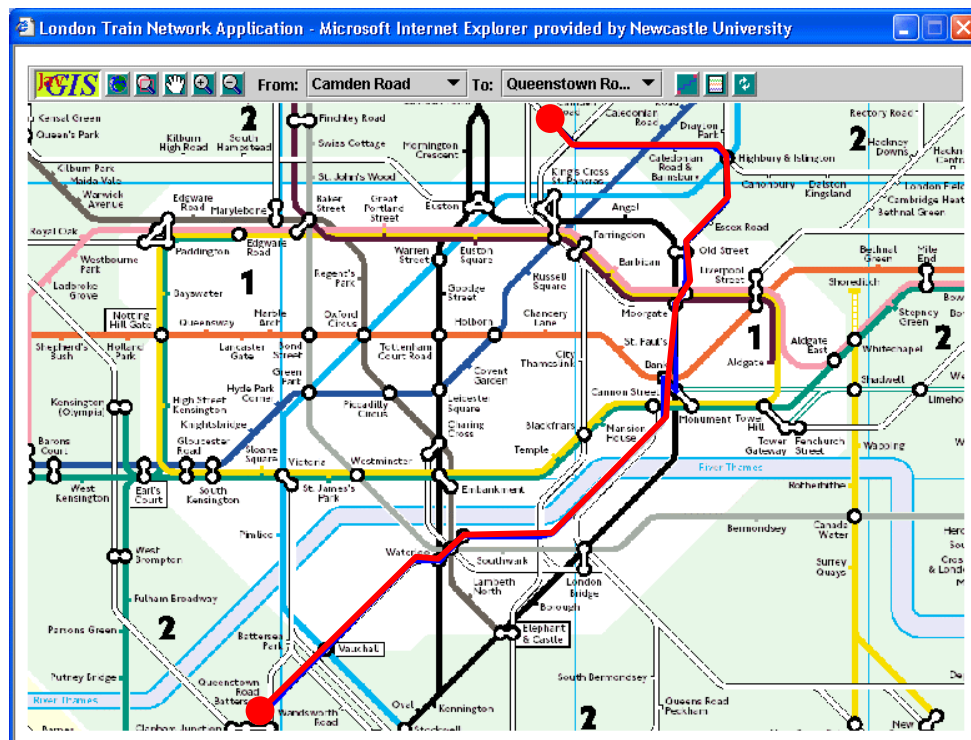


Figure 4. The application system of the London train network application

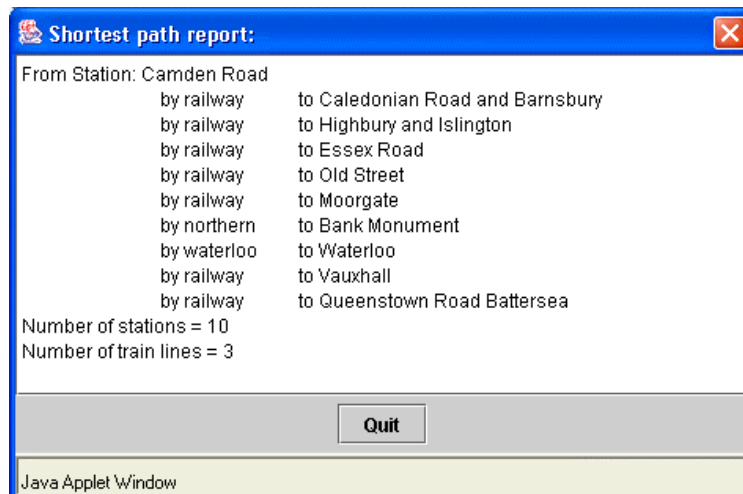


Figure 4. The report of the shortest path

6. Conclusion

This paper has reported the development of the software system that is able to combine data from different data sources through the web and conduct network analysis from those datasets. The advantages of using geospatial interoperable standards and specifications (e.g. GML) have shown the important role interoperability standards have, not only for this implementation but also to the GI community at large. Using GML means that we do not worry about the formats of datasets. Using a web based approach for sharing data maintains data ownership at source but ensures the ability to share that data through various protocols. The system developed shows the possibilities for developing complex analytical GIS applications without regard for proprietary data formats using interoperable standards, interfaces and formats.

Reference:

- Chunithipaisan, S., James, P. and Parker, D. (2002), The integration of spatial datasets for network analysis operations, *Proceedings of MapAsia 2002*, Bangkok, Thailand, 7-9 August,
<http://www.gisdevelopment.net/technology/gis/techgi0065.htm>
- Dijkstra, E. W. (1959), A Note on Two Problems in Connexion with Graphs, *Numerische Mathematik*, vol 1, 269-271.
- ESRI (2002), Geography Matters, *Environmental Systems Research Institute*,
<http://www.esri.com/library/whitepapers/pdfs/geomatte.pdf>, accessed 2 April 2003
- OGC (2001), Web Map Service Implementation Specification, *Open GIS Consortium*,
<http://www.opengis.org/docs/01-068r2.pdf>, accessed 20 May 2004, last updated 27 November
- OGC (2002), Web Feature Service Implementation Specification, *Open GIS Consortium*, <http://www.opengis.org/docs/02-058.pdf>, accessed 20 May 2004, last updated 19 September
- OGC (2003), Open GIS Geography Markup Language (GML) Implementation Specification, *Open GIS Consortium*, <http://www.opengis.org/docs/02-023r4.pdf>, accessed 20 May 2004, last updated 29 January
- Transport (2003a), Fact sheet, *Transportation for London*,
http://www.transportforlondon.gov.uk/tfl/pdffdocs/tfl_factsheets.pdf, accessed 20 April 2004
- Transport (2003b), Tube maps, *Transport for London*,
<http://tube.tfl.gov.uk/content/tubemap/>, accessed 8 December 2002, last updated May