Application of GPRS Technique for Traffic Data Collection System

Communication Software Development and System Evaluation

Wei XU

Faculty of Engineering and Physical Sciences
University of Surrey,
Guildford, United Kingdom
W.Xu@surrey.ac.uk

Mark Phillips

TDC Systems Limited
58 Buckingham Road, Weston Super Mare, United Kingdom
Mark@tdcsystems.co.uk

Abstract— This paper describes software architecture of using GPRS technique to provide wireless data communication for a traffic data collection system. The development of the communication software was based on two scenarios: Centralized Database Management (CDM) and Distributed Data Management (DDM) systems. They were used effectively to improve the current “download on demand” and post event analysis working mode. A “close to the real time” data transmission was achieved. As the essential requirement of an Intelligent Traffic System (ITS), the application of the GPRS technique enabled the user to access the real time traffic data uploaded from multiple road side traffic data collection units. In this study, the authors have successfully developed and tested the communication software of a proposed CDM system and the software architecture of the DDM system was also discussed. For the DDM system, the interface and communication software were designed using Unified Modeling Language and developed as an Object-oriented system based on the .NET platform. The database part was developed using Microsoft SQL. The reliability and the performance of the DDM system were evaluated against proposed hardware configurations.

Keywords—GPRS, traffic data collection, Database, Software Architecture, web services

I. INTRODUCTION

In recent years, the intelligent traffic systems (ITS) [1] have been considered as an effective way to combat fast growing traffic congestion in both developing and developed countries. The front end terminal of the ITS is the traffic data collection hardware which collects vehicle data from the roadside. The data communication between the traffic data collection hardware and user(s) or central database was traditionally achieved via telephone landline or Global System for Mobile communications (GSM) network [1]. However, with increase of the data collection points of the ITS, the day-to-day running cost appeared to be a major challenge, as it became very expensive to use a data transmission scheme charged according to the time connectivity.

GPRS (General Packet Radio Service) is the European telecommunications standard for data transmission used by the GSM network [2]. The advantages quoted by the literature of the GPRS scheme included but not limited to; improved utilization of radio resources, volume-based billing, higher data

transfer rates, shorter access times, and simplification of the access to the packet data networks [3, 4]. These features make the GPRS technique an ideal means of transmitting collected traffic data from multiple road site units to a centralized or distributed database used by an ITS.

GPRS technique also provides an “always-on” mobile communication service and its network usage is based on the volume of transmitted data [3]. The use of GPRS technique for the traffic data collection system can provide all time connectivity and achieve “close to the real time” data transmission while the only charges incurred are when the data is transmitted [5]. This extraordinary feature effectively enables the current stand-alone traffic data collection unit (such as Hi-TAC 100 system supplied worldwide by the TDC Systems) to be connected wirelessly to a Centralized Database Management (CDM) system or a networked Distributed Data Management (DDM).

In this study, the development of the communication software was based on two hardware configuration scenarios: CDM and DDM. With this concept in mind, it became very important to determine how could these resources be shared across multiple users, multiple sites or within a globally distributed traffic company having various roadside vehicle data collection hardware. Those were the key considerations when the communication software and central database were developed.

To take the advantage of the latest Microsoft programming software, VB .NET was used to develop the user interface and GPRS communication software and Microsoft SQL Server 2008 database was used for the development of the traffic database. In order to present the communication software architecture, the system hardware was introduced first based on two scenarios: 1) multiple communication in-stations collecting roadside vehicle data from multiple sites (DDM scenario here after) and 2) a central server computer receiving data from multiple roadside traffic data collect units directly and data was shared by other communication in-stations (CDM scenario here after). In discussion of this paper, the term “in-station” corresponds to the location where a server computer is, the term “site” corresponds to the location on the roadside traffic data collection unit is and central database refers to a database installed on central computer sever.

II. SYSTEM HARDWARE

Figure 1 shows the hardware architecture of the system and associated software. The system hardware consists of multiple

roadside vehicle data collection units, GPRS modem, a central server computer and communication in-station computers.

Figure 1 DTDC hardware architecture

Both CDM and DDM scenarios are shown in the Fig 1. The data flow of the CDM scenario is shown using dark arrows and data flow of the DDM scenario is shown using grey arrows.

III. SOFTWARE ARCHITECTURE

A. CDM software architecture

The proposed CDM software integrated the functions of different hardware, such as data collecting, transmitting, sorting and storing into a unique database to support an ITS. It not only enabled vehicle data from multiple remote sites to be centralized to a SQL database, but also provided web based access to the central database.

Figure 2 shows a block diagram of the software architecture to support the CDM system. The AsyncSocket and multithread functions were used by the communication Server. It enabled the central server computer to receive traffic data from multiple roadside units. The central database was located on the central server computer and SQL server was used to facilitate the large scale data operations of the database used by an ITS.

Figure 2 CDM Software Architecture

To provide online access for the communication in-station users, Web Services technology [7] was used to develop third software running on the central server computer. It enabled the communication in-station users to access the central database using ordinary web browser, such as IE and Firefox, and to carry out data formatting, sorting, classification and statistical traffic data analysis.

B. DDM software architecture

The software of supporting the DDM scenario has been designed using the Unified Modeling Language (UML) [6] to specify the requirements capture and the architecture of the system. It has been developed to execute on a Microsoft Windows .NET platform as a set of business objects that communicate with each other using the Web services technology [7].

UML is a graphical language for visualizing, specifying, constructing and documenting the artifacts of a software-intensive system [6]. In this study, it allowed to cover the conceptual issues such as business processes and system functions. Furthermore, it also provided the coverage of concrete events, such as classes written in a specific object-oriented programming environment (.NET framework), database schemas and reusable software components. DDM architecture was such that, it supported the system being suitably distributed across a network where required for data collection. The function of Web services was also be used to analyze and monitor vehicle data collected from the globally distributed multiple sites [8].

DDM was constructed from a container Multi Document Interface (MDI) shell environment using the .NET Asynchronous Socket technology for effective handling of multiple connections from multiple sites all at once and all within one process / thread. As shown in Fig3, DDM scaled very well as it allows thousands of concurrent connections i.e. only thousands of nonblocking socket objects [9] were used, and not thousands of threads / processes. When a new site made a GPRS connection to an in-station, the DDM Server (within that in-station) raised a socket event (i.e. worker socket) from the thread pool. The responsibility was transferred to the worker socket which in turn has to make sure that the data was collected, formatted, sorted and then stored in a particular user-described location. Once all or some of the vehicle data from a site (for example Site 1) was collected, an appropriate location was automatically selected by DTDC to store all the data from the Site 1. Under no circumstances, the data from one site could be mixed to or confused with the other site, as they were automatically sorted out based on a complex network data sorting algorithm implemented by the traffic data collection equipments.

Figure 3 DDM handling multiple concurrent connections

To provide necessary security of accessing the collected traffic data, users of the DDM must be authenticated to access the data collection system and view the stored data within a distributed environment. When users attempted to log in into the system via DDM shell, their identification details such as username, password, sessions, etc were passed to an authentication function. This function in turn verified the access and loaded the appropriate functionality available for that particular user; from the system registry. Fig 4 is a UML based sequence diagram which shows the flow of control involved when user attempted to login into the DDM

The DDM software is developed using object oriented programming language Microsoft BV .NET. This provides the system with all the advantages of an Object-oriented environment such as inheritance, polymorphism and encapsulation of objects for system reliability and performance. It provides the following system configurations, access and error information, and main traffic data parameters.

- Remote network credentials

Description of the remote connection such as IP address, security identifier, file types, packet data sizes, etc)

- Log details

Activity event details such as logins, file status, remote locations, etc

- Error details:

Failure event details raised during system operation such as socket errors, file creation errors, network errors, etc

- Data size and locations

Description of the vehicle data packets, storage locations, etc

- Server configurations and status

Description of the server environment such as IP address, port details, distributed entity details, etc

- Control functionalities

Loading appropriate views from the system registry based on functionalities

Figure 4 UML-based sequence diagram for login authentication

As it has been described earlier, the DDM supports distribution functionality through web services technology. The ultimate aim of using this technology within DDM is the portability and distribution flexibility obtained when communicating between different sets of business objects. In a distributed environment, the users can configure as many instances they need, and this may lead to multiple instances of that web service to coexist and communicate with each other over the internet using XML based SOAP messages [10].

IV. SYSTEM TEST AND EVALUATIONS

Functional and reliability test of the systems were carried out using UK Vodafone network. Table 1 shows the hardware configuration used to carry out the tests. In this study, only CDM system was test and the DDM software is still in development at the time of this paper was written.

Table 1 Hardware specifications of the system test

	Server PC	TDC unit	GPRS modem	Network
Single unit test	Intel Due core 2.66GHz, RAM, 3GB	EMU [®]	Siemens MC55	Vodafon
Multi unit test	Intel Due core 2.66GHz, RAM, 3GB	EMU [®] and, Hi-TRAC110	TDC GPRS modem	Vodafon

To evaluate the data transmission of a single unit operation, a traffic data simulator was used to provide sensor signals of two lanes to the traffic data collection unit EMU[®] (Event Monitoring Unit provided by TDC Systems Limited). The traffic data simulator was programmed to generate different amount of vehicle data and the EMU[®] was configured to upload its all collected vehicle data once every 5 minutes. EMU[®] was programmed using AT command set [11] to utilize GPRS scheme for vehicle data transmission using Siemens MC55 modem. The Siemens MC55 modem offered the tri-band Class B functionality [12] (i.e. GPRS and GSM, one at a time service) up to multi-slot class 10 data transfer speed (i.e. 16-24 kbps data upload and 32-48 kbps data download maximum) [5].

The Fig 5 shows the structure of the vehicle data outputted by the traffic data collection unit EMU[®].

- Date
- Time
- Serial Number (unique ID number)
- Number of Axles
- Vehicle Classification Index
- Vehicle Category
- Lane Number
- Direction
- Vehicle Straddling
- Validity Code
- Road Surface Temperature
- Individual Axle Weights (ESA) – WIM Version Only
- Gross Vehicle Weight – WIM Version Only
- Gap - Inter-Vehicle Spacing in cms

Figure 5 Vehicle data structure

The reliability and performance assessment of the CDM was divided into two stages. In this context, reliability is defined as the ability of CDM to collect data, and carry out associated functions such as formatting, sorting and then storing it, under given conditions for a given time interval (i.e. one complete day) [13]. The ultimate aim was to avoid losing any vehicle data packets being transmitted from various remote locations, without compromising the performance of the CDM. Performance evaluation corresponds to the delay measured from the time the data packets were collected at an in-station to the time they were processed (i.e. formatted, sorted and then stored accordingly).

A. Stage One

In this stage, CDM was evaluated for reliability and performance under a centralized environment. This means that CDM was implemented as a complete Communication Server to receive vehicle data from multiple sites using GPRS transmission scheme. Table 2 shows that the reliability of the CDM is unquestionable as all the packets transmitted by EMU[®] were collected by the CDM. The result gave one of the samples obtained from a 24 hour test, though the reliability remained consistent throughout the day.

Table 2 Reliability test of the CDM

Test Name: Centralised Reliability Evaluation	
Environment	Centralised
Number of implemented	One
Number of remote sites	Two
GPRS Service Provider	Vodafone, UK
Equipment Used	Siemens MC55 GPRS Modem
ONE SAMPLE	
Data Packets Transmitted X (bytes)	Data Packets Collected Y (bytes)
40	40
150	150
300	300
600	600
1000	1000
1500	1500
$\% \text{ LOSS} = 100 - ((Y - 100) / X)$	
0 (No packets Lost)	

The result of the system performance test is shown in the Table 3. It clearly indicates that the performance of CDM under a centralized environment was very impressive as minimum data packet processing time was observed (on average, 2 seconds processing time throughout). It should be noted that this performance test was for CDM and not for evaluating the performance of the GPRS network. The result shown is one of the samples obtained around midday (11:30am). The performance remained consistent throughout the day.

Table 3 Performance test of the CDM

Test Name: Centralised Performance Evaluation			
Environment	Centralised		
Number of implemented	One		
Number of remote sites	Two		
GPRS Service Provider	Vodafone, UK		
Equipment Used	Siemens MC55 GPRS Modem		
ONE SAMPLE			
Data Packets Transmitted (bytes)	Collection Time (hh:mm:ss)	Processing Time (hh:mm:ss)	Time Delay (sec)
40	11:30:25	11:30:26	1
150	11:32:38	11:32:39	1
300	11:35:41	11:35:42	2
600	11:38:15	11:38:17	2
1000	11:40:05	11:40:08	3
1500	11:42:11	11:42:14	3
On Average, the time delay for DDCS processing was : 2 seconds			

From the above test result, it confirmed that the CDM system developed by the authors has excellent reliability and performance.

B. Stage Two

In this stage, the reliability and performance of the CDM system was tested under multiple connection condition. In such a case the CDM was operating to receive data from multiple traffic data collection units at different site locations using the GPRS transmission scheme. As shown in the table 1, the multiple connection tests were carried out using 2 EMU[®] and 1 Hi-TRAC110 units. Each EMU[®] unit was commended to a traffic data simulator and programmed to upload data every 5 minutes. As there was no more simulator available, the Hi-TRAC110 unit was configured sending nil data package (the data package only had header but no vehicle data) to the CDM every 5 minutes.

The same communication server software was used by the CDM to receive uploaded traffic data from multiple site connections. Table 4 shows the results obtained when three traffic data collection units (2 EMU[®] and 1 Hi-TRAC110) were uploading traffic data using 3 different IP addresses provided by the GPRS modem connect to the units. The data obtained was averaged and shown in Table 4. It clearly shows that reliability is 100% as no data packets were lost.

Table 4 Reliability test of multiple unit connections

Test Name: Distributed Reliability Evaluation	
Environment	Distributed
Number of implemented	Three
Number of remote sites	Three
GPRS Service Provider	Vodafone, UK
Equipment Used	TDC GPRS Modem
ONE SAMPLE	
Data Packets Transmitted X (bytes) for all DDCS	Data Packets Collected Y (bytes) (Average of all DDCS)
40	40
150	150
300	300
600	600
1000	1000
1500	1500
$\% \text{ LOSS} = 100 - ((Y - 100) / X)$	
0 (No packets Lost)	

The performance test of multiple unit connection is shown in the Table 5. It shows that the performance of CDM in a multi-connection condition was excellent as minimum data packet processing time was observed. Compared to the single connection, the multi-connection experienced a performance delay. This delay clearly showed the amount of time taken for CDM to invoke the web services over the network to process data accordingly.

Table 5 Performance test of multiple unit connection

Test Name: Distributed Performance Evaluation			
Environment	Distributed		
Number of implemented	Three		
Number of remote sites	Three		
GPRS Service Provider	Vodafone, UK		
Equipment Used	T D C GPRS Modem		
ONE SAMPLE			
Average of all the data shown:			
Data Packets Transmitted (bytes)	Collection Time (hh:mm:ss)	Processing Time (hh:mm:ss)	Time Delay (sec)
40	11:35:15	11:35:18	3
150	11:36:26	11:36:29	3
300	11:37:12	11:37:17	4
600	11:38:45	11:38:50	5
1000	11:39:35	11:39:41	6
1500	11:41:05	11:41:12	7
On Average, the time delay for processing was : 4.67 seconds = APPROXIMATELY 5 secs			

From the above evaluations, it was clear that CDM reliability was similar in both connection conditions. CDM performance under the multi-connection condition experienced a performance delay on average compared to the single connection condition.

V. CONCLUSION AND FUTURE WORK

The work presented in this paper demonstrated both CDM and DDM systems can provide the reliable and cost effective way of collecting traffic data from remote sites, regardless of location of the sites or distribution of the traffic data collection unit. The use of the GPRS technique in such case not only reduced the running cost, but also achieved “close to real time” traffic data uploading for an ITS.

The test results confirmed the reliable performance of CDM system under both single and multiple connection conditions. A comparison between single and multiple connections of the CDM system showed that reliability was 100% in both cases, as no vehicle data packets were lost when transmitted from the traffic data collection unit to the in-stations using GPRS scheme. Performance of the single connection was understandably better than the multiple connections, as the average data packets processing time for the single connection

was two seconds where three connections was five seconds. This delay was anticipated as communication server invoked over the network, to finally execute the task of processing vehicle data packets from multiple CDM components within the CDM shell. This caused the extra three seconds performance delay.

As the limitation of this study, only the CDM system with single and up to three multiple connections were evaluated. An ITS system would normally consist of hundreds of traffic data collection units. Further test with large number of connection is needed. The implementation of the distributed data management software, as well as the evaluation of the DDM system are also important step forward leading to the full commercial application of the proposed GPRS based traffic data transmission system.

ACKNOWLEDGMENT

Authors would like to thank the funding support from the Knowledge Transfer Partnership program (Partnership No.1146). In addition, the acknowledgement also goes to the supporting staff and engineers of TDC Systems Limited, and Mr Vishal Bartot who worked as KTP Associate for a period of 12 month and produced valuable experiment data for this study.

REFERENCES

- [1] Ralph, D. and C.G. Shephard, SERVICES VIA MOBILITY PORTALS. Future Mobile Networks: 3G and Beyond, 2001.
- [2] Chen, W.Y., J.L.C. Wu, and H.H. Liu, Performance analysis of radio resource allocation in GSM/GPRS networks. Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall. 2002 IEEE 56th, 2002. 3.
- [3] Brasche, G. and B. Walke, Concepts, services, and protocols of the new GSM phase 2+ generalpacket radio service. Communications Magazine, IEEE, 1997. 35(8): p. 94-104.
- [4] Bettstetter, C., H.J. Vogel, and J. Eberspacher, GSM Phase 2+ General Packet Radio Service GPRS: Architecture, Protocols, and Air Interface. IEEE Communications Surveys, 1999. 2(3): p. 2-14.
- [5] McKittrick, D. and J. Dowling, State of the Art Review of Mobile Payment Technology. Retrieved September, 2003. 14(2003): p. 2003-24.
- [6] Booch, G., J. Rumbaugh, and I. Jacobson, The unified modeling language user guide. 1999: Addison-Wesley Reading Mass.
- [7] Schall, D., M. Aiello, and S. Dustdar, Web services on embedded devices. International Journal of Web Information Systems (IJWIS), 2006.
- [8] Casati, F., et al., Business-oriented management of Web services. Communications of the ACM, 2003. 46(10): p. 55-60.
- [9] Welsh, M., D. Culler, and E. Brewer, SEDA: An Architecture for Well-Conditioned, Scalable Internet Services. 2001.
- [10] Gao, B.K., et al., Development of the mobile Web service distribution system platform-independence. Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on, 2004. 1.
- [11] TDC Systems Limited, HI-TRAC EMU Overview, <http://www.tdcsystems.co.uk/downloads.cfm>
- [12] Mc, S.C., MC56 Hardware Interface Description & MC55 AT Command Set & MC55 Release Notes [M],[S. 1.]. Siemens Co, 2005.
- [13] Trivedi, K.S., Probability and statistics with reliability, queuing, and computer science applications. 1982: Prentice-Hall Englewood Cliffs, NJ.