Experiences with a distributed traffic telematics environment – portable travel information system

Andrzej Bargiela[(1)], Evtim Peytchev[(1)], and Richard Berry [(2)]
(1) Department of Computing, The Nottingham Trent University,
Burton St., Nottingham NG1 4BU, U.K.
(2) Nottingham Traffic Control Centre,
Trinity Square, Nottingham, NG1 4AF

Abstract

A distributed software environment that enables modular augmentation of traffic data processing capability in support of traffic/transportation telematics applications is presented. The paper reports experiences with a portable travel information system which illustrates the point that the benefit derived from the UTC infrastructure can be significantly enhanced through distributed applications.

Introduction

The UK Urban Traffic Control Systems have evolved over the last 30 years to cater for the increasing volume of traffic in urban areas. Varying in size from less than twenty signals to large networks with several hundred signals all the traffic control systems share common objectives to reduce delays, congestion and stops. However, with road networks reaching saturation and a difficulty in persuading the commuter to abandon his car in favour of public transport, it is clear that the traffic signal control alone is likely to deliver diminishing returns.

However, the power and versatility of the basic traffic control systems is now being incorporated into an evolving comprehensive Demand Management Toolkit, balancing the need to provide public transport priority and minimisation of environmental impact with the instinctive desire to maximise flow. The basic premise of the on-going development of the urban traffic management and control (UTMC) system in Nottingham is the observation that by linking UTC with other systems such as AVL and passenger information signing schemes it is possible to derive added value from the various single-facetted real-time traffic data. The integration of this data provides a basis for a spectrum of transport telematics applications that can help road users to make the right travel choices on economical, environmental and quality of service grounds.

The ability to share the large amounts of peripheral data generated by ITS depends however on the availability of a suitable computing infrastructure that is capable of evolving so as to reflect the changing traffic/transportation needs of modern cities. The experiences reported here relate to the development, validation and use of such an environment.

Distributed Memory Environment (DIME)

The potential benefits of integrated UTMC systems, providing traffic information and/or supervisory control (using in-vehicle and road-side dynamic route guidance), has created the need for a flexible computing environment in which various new applications can be fully integrated with an existing UTC system without adversely affecting its performance. The best way to satisfy this requirement seems to be the adoption of a distributed computing paradigm. The distributed computers shared memory system (DCSM), developed at NTU, provides such a flexible processing environment for the execution of software modules that process real time data provided by urban traffic control systems and other sources. The system has been called DIME, which stands for a DIstributed Memory Environment.

DIME allows computational tasks to assume a globally shared virtual memory even though the tasks execute on nodes that do not physically share memory. Each DIME application can access global data without having to specify explicitly where this data is or how to obtain it, since any request to access the shared memory is routed by default to the Memory Manager Processor. This is in sharp contrast to message passing systems where each application needs to take care explicitly of inter-process communication, which is a potentially complex and error prone task. DIME provides structured shared memory and it uses sequential consistency model implemented as a centralised memory manager. This design implies that DIME does not require any virtual memory hardware to detect accesses to pages of shared memory and consequently it could be ported to both Unix and DOS environments. DIME system is implemented entirely as a user-level library on top of UNIX and DOS. Unix implementation did not require any kernel modifications because all required communication and memory management functions are standard features of the OS. The DOS implementation of DIME makes use of socket communication libraries from 3COM, [1], which provide full compatibility with the Unix implementation. Programs written in C or C++ are compiled and linked with DIME library using any standard compiler for that language. As a result, the system is relatively portable. DIME's configuration is presented in Figure 1. Each user application code has additional component linked to the code, which provides the communication interface via DIME API with the shared memory system. The requests for reading/writing data from/to the shared memory (creating or removing areas) are transferred by the DIME library over the network to the memory manager task.

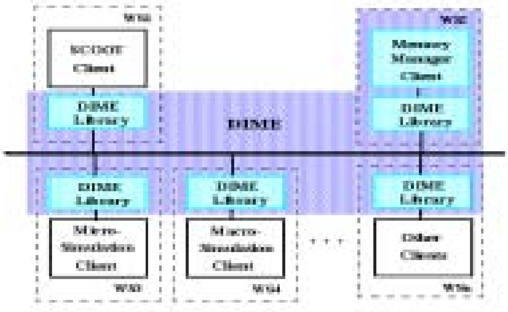


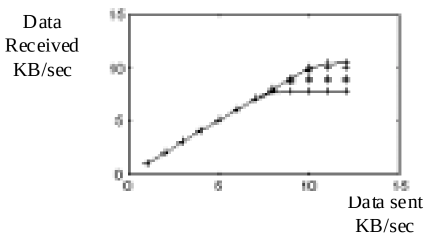
Figure. 1. DIME shared memory system

There are two components of DIME: the shared memory manager (SMM) task which owns the shared data buffers, and the applications program interface DIME-API libraries, which provide a link between user applications and the memory manager. The SMM component of DIME operates on a closed-loop basis, continually checking the requests to access or to maintain the shared memory data structures. The DIME's API provides facilities for creation and removal of shared memory structures and the read/write access to them. The synchronisation of access to shared memory is implicit in the design of the memory manager task so that the application programs do not need to concern themselves with the complexities of data synchronisation in a distributed computing environment.

The DIME system has been deployed in the Nottingham Traffic Control Centre since 1997 and it has been extensively evaluated using both local- (LAN) and wide-area (WAN) networks. Performance tests were carried out for the "worst case" scenario of executing the memory manager on a remote computer abroad (in Finland) while the application tasks were situated on nodes at NTU. Consequently, every access to shared memory required network transmission. The "writer" task wrote data into the shared memory with a progressively increasing rate and the "reader" tasks attempted to keep-up with the "writer", retrieving all the data. The experiment was repeated with 1 to 4 readers. The experiments showed that in WAN working environment the maximum information retrieval rate was in the range 2.8-6.5 KB/sec. This performance did not depend substantially on the number of "readers" but it did depend on the network load. The maximum retrieval rate of 6.5 KB/sec was measured with 1 to 4 "readers" on a lightly loaded network and similarly the maximum rate of 2.8 KB/sec was measured with 1 to 4 readers with a busy network.

Another set of tests was carried out in a LAN working environment. Using the same suite of application tasks the performance of DIME has been found to be in the range 8-10.6 KB/sec. Over 99% of the time were spent on negotiating message delivery by TCP/IP and the remaining 1% on the actual shared memory accesses and housekeeping.

Figures 2 and 3 depict the respective performance envelopes for the LAN and WAN tests of the system. The tests confirm that the WAN configuration offers an adequate distributed traffic systems software development environment. However, because of the wide fluctuations of the data traffic over WANs, for the real-time operation of the system the DIME software should execute using LAN configuration, as originally intended. It must be pointed out however that this is not a limitation on the part of DIME and the use of more efficient WAN networks may well alter this conclusion.



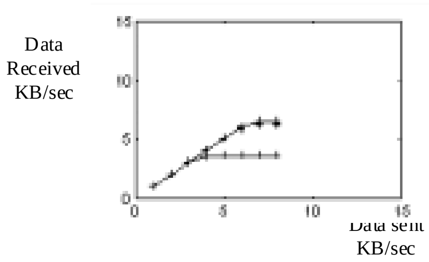


Figure 2. LAN performance of DIME (2-5 clients) **Figure 3**. WAN performance of DIME (2-5 clients)

The DIME software has been designed to support a wide spectrum of traffic and transportation telematics applications through the provision of customisable static and dynamic data structures and the flexible applications programming interface. An overview of applications, that are interfaced to SCOOT via DIME, is given in [Bargiela and Berry, 1999]. These include the following: a predictive macroscopic simulator [Peytchev et al. 1996]; a real-time microscopic traffic simulator [Kosonen et al. 1998]; and a geographical information system (GIS) based interface to UTC [Claramunt et al. 1999]. This paper reports experience with another application interfaced through the DIME system, that of a portable traffic/travel information.

Portable traffic/travel information system

The idea behind traffic and travel information systems is to assist drivers with the selection of routes according to the current urban traffic situation and, very importantly, to inform the actual and potential drivers about the alternative modes of transport. It is clear that the integration of data concerning traffic flows, public transport and individual journey plans makes it possible to perform multi-modal optimisation of travel. However, the challenge was to develop a system capable of simultaneous data acquisition, processing and dissemination of the traffic/travel advice in real time to a full spectrum of end users.

Traveller information solutions, aimed at WWW-users, have been developed by researchers in UK, Europe and USA as surveyed in [Padmos and Bernstein, 1997]. A representative of this class of solutions is the ROMANSE system which provides a multi-modal urban information and journey planning [Romanse-on-line, 1997]. The system is expected to be accessed from home/office and/or travel information terminals situated in the strategic places in the city. Early evaluations of the system by the developers suggest significant impact of the system on the planning of non-commuting journeys by the WWW-users. However, the dynamic nature of urban traffic and a relatively small proportion of travellers that make use of WWW, limits the accuracy and the relevance of the advice offered by such a system. At the other end of the spectrum, travel information systems incorporated in top-of-the-range cars, offer excellent static travel information but are rather limited regarding the real-time traffic and public transport data provision and as such do not contribute significantly to mode switching decisions by drivers.

The portable traffic and travel information system developed at NTU combines the functionality of both ‘in-car’ and ‘road-side’ information systems and it is implemented using the handheld electronic organisers (PSION) and the GSM telephone. Since both pieces of equipment are standard consumer electronics products, which are not constrained by being a part of the car’s equipment or being deployed at a particular road-side location, the system is expected to have a much broader user base and, by implication, a much greater impact on mode switching decisions.

The overall design of the system is presented in Figure 4. Each mobile application environment consists of two parts: the application server which is interfaced to the DIME environment and the mobile client which provides communications and user interface to the server. The mobile client is typically configured as both information recipient and information provider (e.g. information about the intended journey) thus contributing to the enrichment of the information content of data processed by the server. The integration of the portable information system within the framework of the distributed memory environment (DIME) facilitates also real-time access to traffic data and the computational results by all other applications in the DIME environment. In particular, access to information on predicted journey times and the associated confidence limits of such predictions is expected to provide a valuable reference for mode switching decisions.

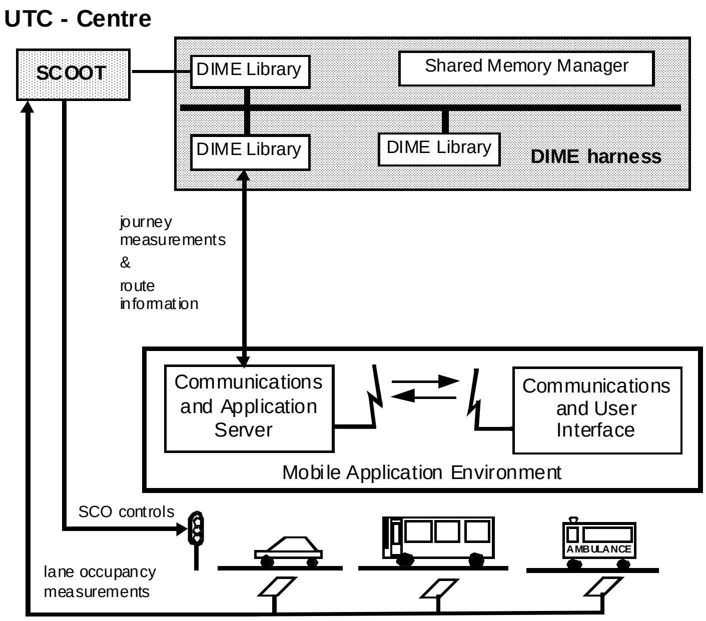


Figure 4. DIME interface to a portable traffic/travel information system

The GSM phone connection between the mobile client and server can be accomplished using either GSM voice- or signalling-channel. The voice-channel communication option has been investigated in the SOCRATES project [Socrates, 1992] and it has been criticised on the grounds of unacceptably long delays when negotiating the point-to-point protocol and the high cost of the continuous connection to the server. Similar conclusion has been reached more recently by other researchers [Lyons et al., 1997, Padmos et al., 1997] in the context of the GSM-based WWW access. The second alternative, that of using the signalling-channel, was deemed to be more appropriate (in that it affords asynchronous mode of operation) and it has been adopted in our system. Since the GSM signalling messages are buffered by the communication network itself, the total connection time of individual users is of the order of few seconds, thus falling well within the acceptable delay. However, this mode of operation is severely constrained by the maximum number of bytes in data packets. In order to overcome this difficulty our mobile client application has been designed for a PSION range of PDAs which offer not only a powerful message decoding tool but also the possibility of local storage of semi-static data and the flexibility of graphics user interface (Figure 5).

Figure 5. ‘In-car’ execution of a mobile client on PSION WorkAbout

One of the major criticisms of the current demand responsive UTC systems is that they react to the already existing traffic situation rather than anticipating the onset of traffic congestion and trying to prevent it. This is primarily because it is both hard to predict the intentions of road users and, more importantly, to assess the accuracy of any such prediction so that one may have a degree of confidence that any supervisory traffic control will not be counterproductive. Consequently, in order to be credible, the traffic and travel information system, which can be seen as a form of supervisory traffic control, needs to recognise and quantify the uncertainties associated with both traffic measurements and the underlying traffic model. Using our Confidence Limit Analysis methodology, [Peytchev, 1999], we have shown that for any metering data (given meter position and accuracy) it is possible to calculate error bounds (confidence limits) on the travel time predictions. This result can be used in two ways: firstly it can be used to provide a more realistic advice about journey times, which recognises the statistical variability of driving styles and traffic conditions; and secondly it can be used to assess the effectiveness of various measurements in a traffic telemetry system.

Figures 6 and 7 present sample graphical user interfaces to the traffic/travel information system implemented on PC and PSION WorkAbout platforms respectively. The journey time predictions are accompanied by the confidence limits envelopes. It is interesting to note that the variation of travel time is considered by most drivers as a more critical piece of information than the travel time itself and appears to be a principal factor in persuading drivers to switch to public transport.

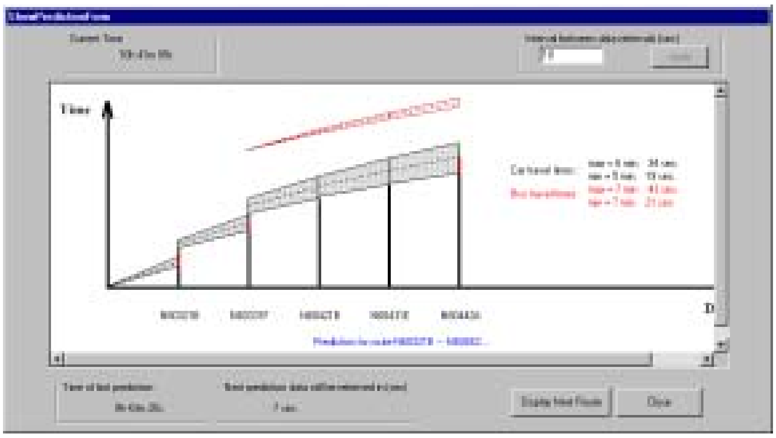


Figure 6. PC-based interface to the mobile traffic/travel information client

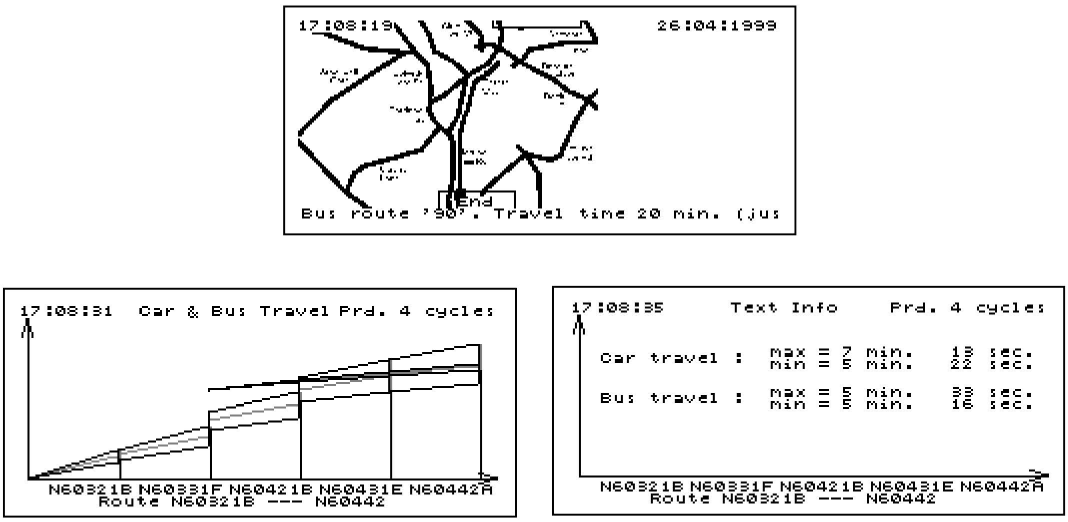


Figure 7. PSION-based interface to the mobile traffic/travel information client

Conclusions

The initial motivation behind the development of the DIME system was a desire to make better use of the data generated by the UTC/SCOOT system without having to enter into major UTC software changes to develop new facilities. DIME has successfully demonstrated the potential for added system value without the necessity for such software changes and without imposing undue overheads on the host system. The system design is such that any future ITS facility can easily be added to the DIME environment and therefore provide an accessible path forward from Traffic Control System to Demand Management Toolkit.

The set of 4 distributed applications trialled at NTU/NTC illustrate the point that the benefit already derived from the telemetry/telecontrol infrastructure of any UTC system can be significantly enhanced by complementary applications. In particular, the portable traffic and travel information system, described in this paper, offers a potential of making a significant contribution to the enhancement of sustainability of cities through better route choice and/or persuading drivers to use public transport.

References

3COM, TCP/IP with Demand Protocol, *Architecture Programmer's Reference manual*, Part No 8216-00, December 1990.

Argile A., Peytchev, E., Bargiela, A. and Kosonen, I. (1996) DIME: A Shared Memory Environment for Distributed Simulation, Monitoring and Control of Urban Traffic. In *Proceedings of European Simulation Symposium ESS'96*, Genoa, vol. 1, pp. 152-156.

Bargiela A, and Berry, R. (1999) Enhancing the benefits of UTC through distributed applications, *Traffic Technology International,* Feb./March, pp. 63-66.

Claramunt, C., Peytchev E., Bargiela A., (1999) A real-time GIS for the analysis of traffic system, Proc. of IEEE Conference, Sept. '99.

Claramunt, C. and Thériault, M. (1996) Toward Semantics for Modelling Spatio-temporal Processes within GIS. In *Advances in GIS Research I*, M. J. Kraak and M. Molenaar (eds.), Taylor and Francis, pp. 27-43.

Etches, A., Claramunt, C., Bargiela, A. and Kosonen, I. (1999) An Inter-operable TGIS Model for Traffic Systems. *In Innovations in GIS VI*, Gittings, B. (ed.), Taylor & Francis, in press.

Kosonen, I. and Pursula, M. (1995) Object-oriented and Rule-based modelling - Experiences from Traffic Signal Simulation. In *Proceedings of the 6th International Conference on Computing in Civil and Building Engineering*, Berlin, Germany.

Kosonen, I., Bargiela, A. and Claramunt, C. (1998) A Distributed Information System for Traffic Control. In *Proceedings of the Xth European Symposium in Simulation Systems*, Nottingham, A. Bargiela and E. Kerckhoffs (eds.), pp. 355-361.

Lyons G., et al., (1998), Traveller information and the Internet, Traffic Engineering and Control, Vol. 39, No. 1, pp.24-31.

Padmos J., Bernstein D., (1997), Personal travel assistants and the World Wide Web, Paper no. 970621, The Transportation Research Board, 76th Annual Meeting, Jan. 1997.

Peytchev, E., Bargiela, A. and Gessing, R. (1996) A Predictive Macroscopic City Traffic Flows Simulation Model. In *Proceedings of European Simulation Symposium ESS'96*, Genoa, vol. 2, pp. 38-42.

Peytchev E. (1999) *Integrative Framework for Urban Traffic Simulation and Prediction and Confidence Limit Analysis of the Predicted Traffic Flows*, Ph.D. thesis, Department of Computing, The Nottingham Trent University, Nottingham, UK.

Pursula, M. (1998) Simulation of Traffic Systems: An Overview. In *Proceedings of the Xth European Symposium in Simulation Systems*, Nottingham, A. Bargiela and E. Kerckhoffs (eds), pp. 20-24.

ROMANSE ON-LINE, (1997), URL: http://www.soton.ac.uk/~trgwww/

SOCRATES KERNEL, (1992), DRIVE II Project V2013, European Comission, Dynamic Route Guidance System, Report 03.43.92

SMARTEST (1997), Review of micro-simulation models, Project Deliverable D3, EU Framework 4, DG VII.