Building software based intelligent transport system: A storage networks analogy

Aravind C V[1] Dr N K Cauvery[2]
[1]Student, Department of Information Science and Engineering,
[2]Head of Department, Department of Information Science and Engineering,
[1, 2] RV College of Engineering, Bangalore

Abstract--- Access to Real time information on the services provided by a transport operator or different transport operators is the primary requirement of all the users of the transport system. This helps the passengers to attain zero waiting time and also plan their journey well in advance. This paper describes how to build a computer system, conceived as an intelligent system for public transport management. The goal of this system is to help users in planning their trip from a source location to a destination point. When building such a system we can note that its architecture is very similar to that of a storage network where the data migration takes place between a source storage system and a target storage system. Hence we can take this analogy to build an efficient intelligent transport system.

Keywords: - SAN, Public transport system, Data Migration, hosts, switches, storage arrays.

I. INTRODUCTION

Public mass transit systems are gaining popularity worldwide due to increased traffic congestion, urbanization, and changes in population density, particularly in urban areas. Increase in travel time, air pollution, and fuel consumption, is causing various social, environmental, and economic problems. Due to these reasons majority of the urban population prefer to travel in public buses, trains and other passenger transport systems.

To enhance the commuter experience it is essential to have an integrated approach for modelling transport infrastructure and optimizing passenger transport in urban areas [1].

This paper focuses on building a software based intelligent transport management system that assists commuters to plan their trip from a source to a destination location in an urban network.

II. MODELING THE SYSTEM

A public transport network can be modelled similar to that of a storage network within a SAN environment. At the most basic level a storage area network consists of servers and storages that support FC (Fibre Channel) technology. Additional components like routers, multiplexers, extenders and gateways can be used in a SAN. Each of these components are interconnected using the high speed links like Fibre Channel (FC) or Small Computer System Interface (SCSI) [2] [3]. Figure I show the elements of a typical storage are network.

The idea of a commuter using the public transport to travel from one location to another is similar to the migration of data from one storage system to another over a SAN using agentless monitoring. Agentless data collection involves collecting data from computers without installing any new agents on them. An agent is a software program (sometimes called a service or daemon) that runs on a computer with the primary purpose of collecting information and pushing it over the network to a central location. The advantage of using an agentless system is that they lower the cost of ownership, reduce management overhead, and provide for quick and easy deployment.

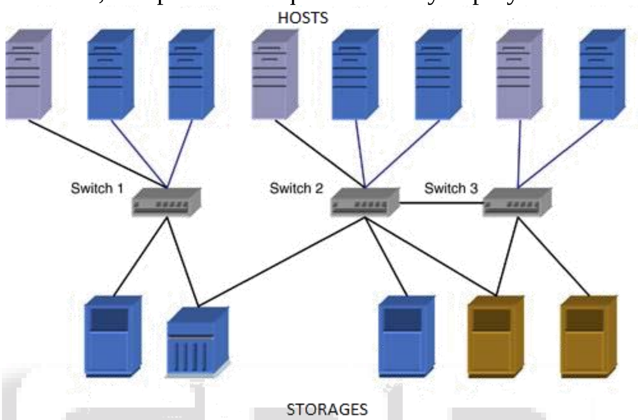

Fig. 1: A typical Storage Area Network

A data migration tool in an agentless environment typically offer the following services before the actual migration begins: (A) Discovery of components (B) Selection of source and target storages among those present in the inventory (C) Analysis of selected components and (D) Planning of the migration activity [4]. The coming sections of this paper clearly explain how the above activities can be incorporated in the intelligent public transport environment.

A. *Discovering the Components*

Discovery of various components is the starting point through which all the information required to implement this system is gathered. All the other following activities are dependent upon the effectiveness of the discovery process. Discovery in an agentless environment is the process of retrieving relevant information from various sources into a central repository. The information that are discovered in a typical storage environment can be ipaddress, host- switch and storage details, operating system details, connectivity details, link speed etc. Similarly, information like geographical locations, bus and train stations, roads and train connectivity details etc. can be discovered from different servers, in a public transport environment.

Discovery can be performed in two ways, i.e. static discovery and dynamic discovery. Static discovery can be used to gather information from sources that do not vary greatly with respect to time. Hence, data like geographical details, bus or train station details can be retrieved using static discovery process. Static discovery is usually a one-time activity. Those components that are statically discovered are stored permanently in the inventory.

On the other hand, dynamic discovery is used to retrieve data that is constantly changing. Information like regular traffic updates, bus and train schedules, GPS information etc. can be retrieved using dynamic discovery. Dynamic discovery can be achieved in different ways – (A) regular polling by the central server to the information servers (B) Information servers pushing the data into the central server at regular intervals of time or (C) Central server pulls data from information server on the need basis. Figure II shows the types of dynamic discovery of components.

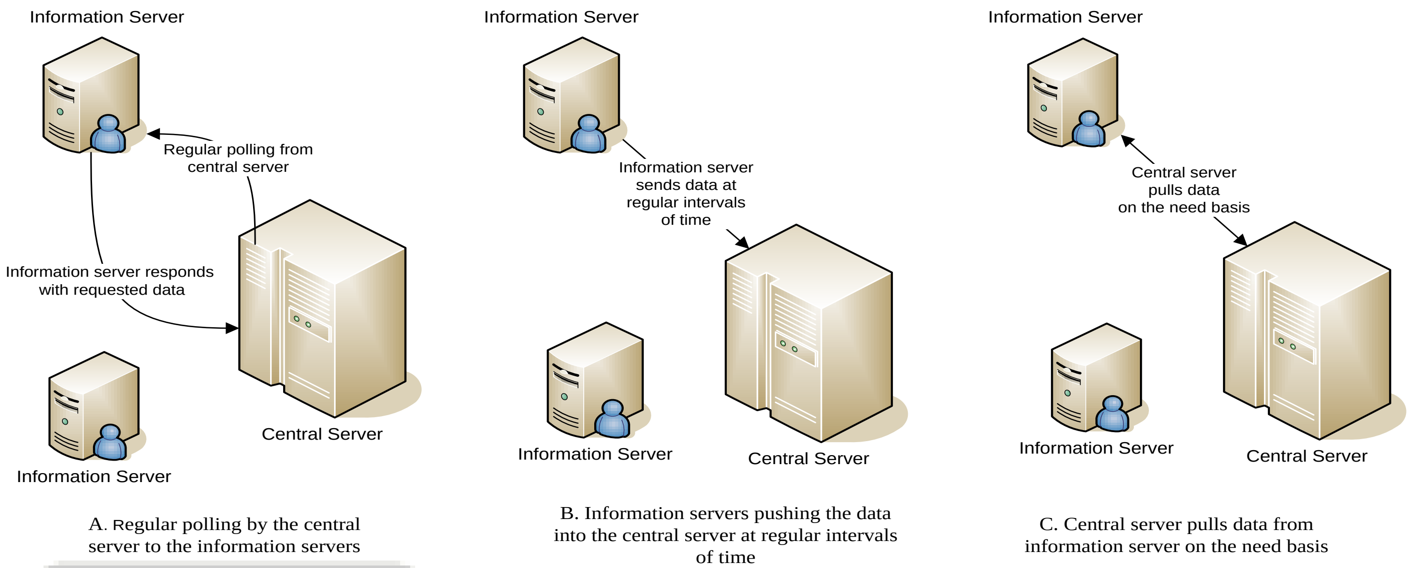

Fig. 2: Types of Dynamic Discovery of components

B. Storing Details in Inventory

Inventory is the repository of all information needed in implementing this system. All the information discovered in the previous step is stored in the inventory. Some existing data such as maps representing geographical information, existing transport schedules may be directly imported into the inventory bypassing the discovery step. In a typical SAN environment, inventory consists of details like component name, operating system, port information, connectivity details etc. Inventory is the starting point through which all kinds of analytics on data can be performed.

C. Selection of Source and Destination from Inventory

After discovering the required details and storing them into the inventory, the source location and the destination of the commute have to be selected from the inventory. Selecting the source and destination points can be quite a challenge to the user. To assist the user in the selection of the source and destination, a topological diagram of the discovered components will be useful. A topological diagram is a diagrammatic representation of the inventory with all the components represented as nodes and links.

Selection of source and destination points should automatically select all components that are related to these points. For example: all the in-transit points between the source and target commute locations are automatically selected in a public transport network. In such a scenario there may be more than one route that may be selected between the source and the target.

D. Analysis Based On Selection

Once the source and the target points of commute are fixed, the various routes that lie in between the source and target end points have to be analysed to select the best route or the best mode of transport. Analysis can be performed based on various parameters such as time taken for travel, cost of travel, mode of transport etc. Apart from these parameters, various other constraints like connectivity failures, traffic bottlenecks, transport schedules etc. have to be considered while performing analysis.

Analysis in SAN environment includes checking the compatibility and configuration of components like hosts, switches and host bus adapters with respect to the target storage array. Analysis also considers information like port speed details, network link connectivity failures and network traffic to determine the routing path from source array to target for data migration.

E. Planning

Planning is the phase which helps to plan and schedule the commute based on the analysed data. Planning helps in deciding on the route to take, mode of transport, expected fare of commute and expected time that would be taken to reach the destination. The past experience of the commuter and the previous data of transport and connectivity help in planning the journey. Big data analytics can be used to efficiently plan a commute as it gives accurate data based on specific patterns. Planning a journey from a specific source to destination may differ from commuter to commuter based on their specific needs.

Figure 3 shows the routes obtained from analysis for a source and destination at peak hour and non-peak hour times. In the above figure, the dotted arrows indicate the routing recommendations at 7 AM when the traffic density is less. The normal arrows indicate the routing recommendations at 9 AM when the traffic density would be high on the route.

Planning in SAN environment generally refers to recommendations to update the drivers or firmware versions to the latest versions of the software or planning to migrate a specific set of data to the destination storage array.

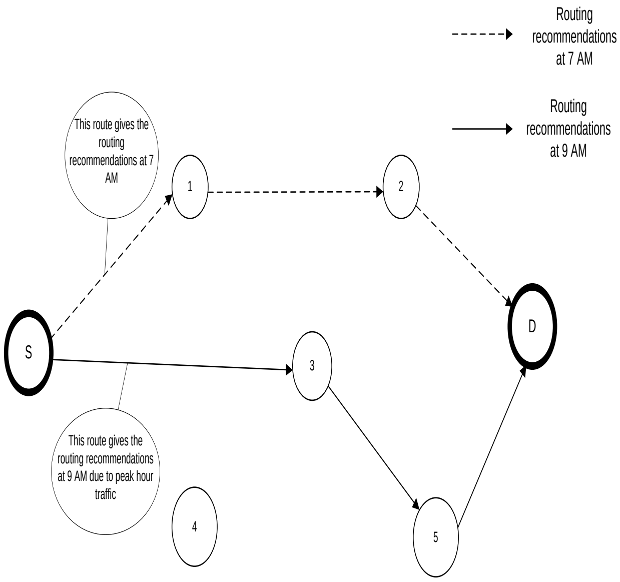

Fig. 3: Routing recommendations at peak hour vs. non-peak hour traffic

F. Execution

Execution is the final stage where the commuter actually starts the journey from source and reaches the destination as per the plan. It is important to note that at times the journey may not take shape as per the planning done in the previous step. In such cases, the overall analysis done on all possible routes and modes of transport will be handy in completing the journey.

Similar to the public transport networks, execution in storage networks refers to the actual migration of data between the source storage array and the destination storage array.

III. Related Works

Bazzan et al. [5] suggests that the transport domain deals mainly with people moving from one place to another in an urban environment. This is a very dynamic domain because of the interactions involved between the various players, for instance pedestrians, cars, buses, trains and cabs among others. Additionally, the infrastructure is also to be considered like streets and sensors underneath, traffic lights, and electronic signs.

The fluctuations in the transport demand during a period of time like unexpected accidents and faults, delays, energy consumption, and a limited transport supply make this domain increasingly complex and calls for the need of approaches to distribute the demand within the infrastructure and optimise its usage [5]. Existing approaches focus on diverse and specific aspects of transport, for instance on traffic coordination at traffic management centres, cf. [6], maximisation of (urban) network throughput by controlling the signalling at adjacent road intersections, cf. [7], or improvement of the traffic flow by optimising both the timing of traffic lights and routes selected by drivers, cf. [5]. Software agents and multi-agent systems [8] are increasingly being used to solve overarching issues in the domain of traffic and transportation systems, namely traffic congestions, vehicle emissions, and transportation coordination. The applications of agent computing paradigm ranged from modeling and simulating traffic, managing congestion, and dynamic routing and recommendations [9]. Urban traffic control strategies to reduce traffic jams include intersection signal control [10], bus fleet management, integration of urban traffic control and route guidance, and intelligent route guidance.

Agent-based platforms for road traffic management include TRACK-R for route guidance [11], MAS incident manager for incident management [12], Mobile-C for uncertainty handling [13], and aDAPTS [14] for traffic congestion management. These platforms use agents to model the behavior of vehicles, signal lights, and road segments. The majority of these agent-based architectures model only a single mode of transport and do not account for the diverse needs of commuters. However, the architecture described in this paper endeavours to fulfil this gap by gathering data from all common modes of transport in urban areas, relying on real time traffic data and traffic forecasts, to create a richer and multi-layer transport network that is capable of devising flexible route guidance.

IV. Conclusion

In summary, the model described in this paper can be an adequate solution to construct an intelligent software system for public transport management. The model is a global solution that integrates different types of methods and problem-solvers to support four basic tasks: discovery, selection, analysis and planning. The model described in this paper is analogous to the data migration model generally followed in SAN networks. The architecture is innovative in the sense it covers various modes of transport including road, public transport, trains, cycling routes, and pedestrian paths. It also uses various measures to calculate the cost of using each segment. The outputs are intermodal route solutions that accommodate commuters' preferences and demands such as time of travel and possible modes of travel.

References

[1] Wei-Hsun Lee, 2011. Discovering Traffic Bottlenecks in an Urban Network by Spatiotemporal data mining on location-based services. IEEE Transactions on Intelligent Transportation Systems, VOL. 12, NO. 4, DECEMBER 2011.
[2] Enrico Schiattarella, "Introduction to Storage Area Networks," Cisco Systems, October, 2002, http://www.telematica.polito.it/oldsite/protocolli/schiattarella.pdf
[3] Karl L. Swartz, "HP 3PAR Fast RAID: high performance without compromise," Hewlett-Packard Development Company, L.P., 4AA3-4815ENW, June 2011.
[4] Technical White Paper, "HP Data Migration Services," Hewlett-Packard Development Company, L.P., 4 AA3-0774ENW, August 2010.
[5] Bazzan, A.L.C., de Oliveira, D., Klügl, F., Nagel, K.: To adapt or not to adapt – consequences of adapting driver and traffic light agents. In: Tuyls, K., Nowe, A., Guessoum, Z., Kudenko, D. (eds.) Adaptive Agents and Multi-Agent Systems III. LNCS (LNAI), vol. 4865, pp. 1–14. Springer, Heidelberg (2008)
[6] Hernandez, J.Z., Ossowski, S., Garcıa-Serrano, A.: Multiagent architectures for intelligent traffic

management systems. Transportation Research Part C: Emerging Technologies 10(5-6), 473–506 (2002)
[7] Basile, F., Carbone, C., Chiacchio, P., Boel, R.K., Avram, C.C.: A hybrid model for urban traffic control. In: 2004 IEEE International Conference on Systems, Man and Cybernetics, vol. 2, pp. 1795–1800 (2004)
[8] Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. Autonomous Agents and Multi-Agent Systems 11(3), 387–434 (2005)
[9] Chen, B., Cheng, H.H.: A Review of the Applications of Agent Technology in Traffic and Transportation Systems. IEEE Transactions on Intelligent Transportation Systems 11(2), 485–497 (2010)
[10] Roozemond, D.A.: Using intelligent agents for pro-active, real-time urban intersection control. European Journal of Operational Research 131(2), 1, 293–301 (2001)
[11] Garcia-Serrano, A.M., Vioque, D.T., Carbone, F., Mendez, V.D.: FIPA-compliant MAS development for road traffic management with a knowledge-based approach: The TRAKC-R agents. In: Proceedings of Challenges Open Agent Systems Workshop, Melbourne, Australia (2003)
[12] Tomás, V.R., García, L.A.: Agent-Based management of non-urban road meteorological incidents. In: Pěchouček, M., Petta, P., Varga, L.Z. (eds.) CEEMAS 2005. LNCS (LNAI), vol. 3690, pp. 213–222. Springer, Heidelberg (2005)
[13] Chen, B., Cheng, H.H., Palen, J.: Integrating mobile agent technology with multi-agent systems for distributed traffic detection and management systems. Transportation Research Part C: Emerging Technologies 17(1), 1–10 (2009)
[14] Wang, F.Y.: Agent-based control for networked traffic management systems. IEEE Intelligent Systems 20(5), 92–96 (2005)