Architecture Design and Prototyping of Front-End Component For Integrated Transportation System

Waskitho Wibisono[1], Hideto Ikeda[2], Nikolaos Vogiatzis[3]
[1] Department of Informatics, Sepuluh Nopember Institute of Technology(ITS) ,Indonesia
[2] Cyberspace Laboratory, Ritsumeikan University, Japan
[3] Transport Systems Centre, University of South Australia, Adelaide Australia.

1 Introduction

Our societies heavily rely on transportation systems. The usage of the transportation systems is greater than ever before. Moreover, the volume of the system has been increasing rapidly in not only developed country but also developing country. According to those phenomena, our laboratory organized a project of integrated transportation system order to improve efficiency and safety of transportation system namely “PROJECT for FUTURE INTEGRATED TRANSPORTATION SYSTEM”.

The missions of the project are intended to improve efficiency and safety of transportation systems on traffic control and monitoring, transportation planning, facilities and resource management, and decision support system of the transportation systems. On the other hand, in order to develop and construct such huge multi-missions, multi-technology and organizational systems, it is necessary that a top-down approach and comprehensive system design methodology be established which will make the implementation of such a huge systems more realistic and implement able.

We have proposed Three Layers Object Model (3LOM) as a top-down architecture design approach to implement the integrated transportation system (Ikeda *Et al.,l* 2003) which front-end component is bottom layer of the 3LOM architecture. The Figure 1 shows the 3LOM architecture design

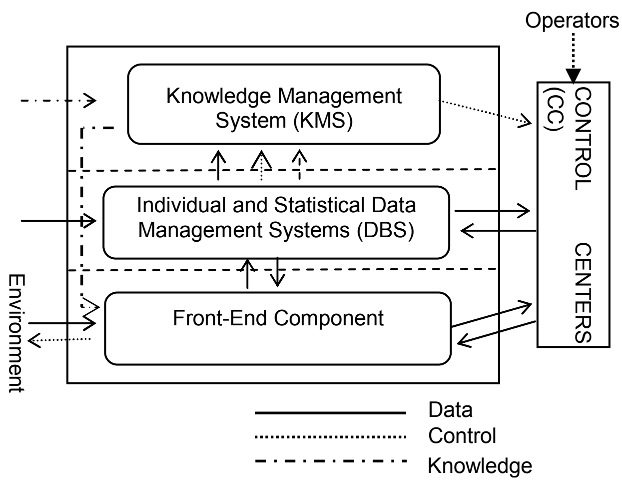

Figure 1. 3LOM architecture

2 Background

Currently there are many existing transportation systems from road-based traffic control systems, train-control system and automobile navigation system. In addition, other part of the systems such as facilities management and planning system play important roles to support the entire systems. In facts, most of those parts nowadays work independently and not to be combined as an integrated system yet.

Most of recent transportation control systems nowadays are centralized system. Using those systems above, all data need to be transferred to the central computer, subsequently the central computer needs to create decisions and distribute those to each control device back. Those approaches will increase response times as well as network bandwidth consumption to send data from all sensors and to distribute information back again to each related location. Moreover, from the view of technology, complexity and scale of these systems have made the difficulty to implement integrated system idea. Another important factors are most of those systems were start to be developed when technology for data communication and integration not as advance as nowadays.

Those phenomena have motivated us to propose new approach to design and implement an integrated transportation system which focus not only in traffic monitoring control but also other components of transportation systems such us facilities and resource management, planning systems using decentralized system approach. For developing such a huge integrated system for the transportation systems, we need some strategies. It is not enough by only providing the architecture of distributed system but we need more integrated approaches since the development processes will involve many responsible organizations, developers, private sectors and so on.

We consider that to develop the integrated system we need high flexibility and interoperability among devices and systems to enable integration with various independent systems. We need also secure and robust integrated system supports. Moreover, scalability also very important which allows us to expand or to integrate the system. We need standard for data structure and communication for the integration purposes.

These things become motivations of this paper to propose a standard component within 3LOM architecture layers and how it can be integrated with other layers or control centres with high flexibility to increase efficiency of development cost.This paper proposes an architecture design and prototyping of the front-end component that is the bottom layer of the 3LOM architecture. Furthermore, we show how this component works with other layers and control centres in the system.

3 Front-end component in the integrated transportation system

In the 3LOM, the front-end component is the bottom layer which plays roles to manage specific localized locations or environment. It acquires data from those locations and performs independent controlling mechanism for its localized environment. Some data is used specifically for the control of individual locations whereas other data is locally summarized and then transmitted to the middle layer as the data management layer

This paper concentrates on the architecture design of the front-end component as a new type control unit for the proposed integrated transportation system. From an implementation perspective of the component; we need to devise an optimum strategy to integrate them into a networked real-time and distributed system.

The most important of all is the need to establish an appropriate strategy as to standardize the component which allows generic unit controllers to be tailored for individual required functions (W.Wibisono Et all., 2004). We analyse requirements of this component and describe detail information of the component as follows:

3.1 Multi-purposes

First characteristic of the front-end component is multi-purpose. As the integrated transportation systems are large systems we need a standard distributed object in order to get higher flexibility and expandability of the systems.

Front-end component should be a standard device which can be customized according to its role in a localized environment. Each of front-end components in each location will probably have different characteristics and local mechanisms however the functions of those front-end components are same in general.

3.2 Distributed component

As an element of distributed systems, a front-end component is a distributed component also. Consequently, each of the front-end components should have capabilities to work independently and perform its local functions to manage the environment without need to be managed by the control centres.

Front-end component should be an autonomous and intelligence object inside the 3LOM with high independency and responsibility of its controlling area by using its local knowledge and intelligence system

3.3 Flexibility

Nowadays there are many devices in transportation systems. In the front-end device, data from environment are acquired by using various sensor devices form many vendors and also for delivering action to the environment also use many types of actuators from different manufactures.

In order to have high flexibility to handle many devices, front end components should be implemented as open systems which enable to all of the manufactures to refer the specification and create devices which provide high interoperability among many devices from different manufacturers.
Flexibility is one of important requirements to achieve lower cost for implementation and maintenance of the systems.

3.4 Scalability

As a distributed component inside distributed systems, the front-end component will be implemented as multipurpose devices which have capability to be customized according to its localized environment.

Scalability is to be understood as the possibilities to extend the system to achieve the appropriate purposes by integration of the local unit with other local units and control centres.

3.5 Robustness

As an outdoor device, there are many factors we need to consider as other significant aspects. Robustness aspects are related with dependability of the component related with performance of the component in any condition caused by environmental problem or communication problem.

We need to design the device and consider about the standard outdoor device standard in answer requirement in a hostile environment, such as traffic signals and traffic management controllers need to be a robust device. Other important aspects are the robustness of the system performance.

There are many conditions which will affect to performance of system, for example such as communication problem, sensor damage, broken actuator and any specific problems which affect directly to the performance of the system etc. n case of any sensor or actuator damage the front-end component need to report to any related control centre and make self adjustment of the condition.

4 Architecture of the front-end component

The Figure 2 shows the architecture of the front-end component. It defines structure of functional objects of the front-end component which consists of 9 objects. The architecture shows how the relations and coordination among all of those objects are designed. This component architecture is designed in order to attain the requirements we have defined in the previous section.

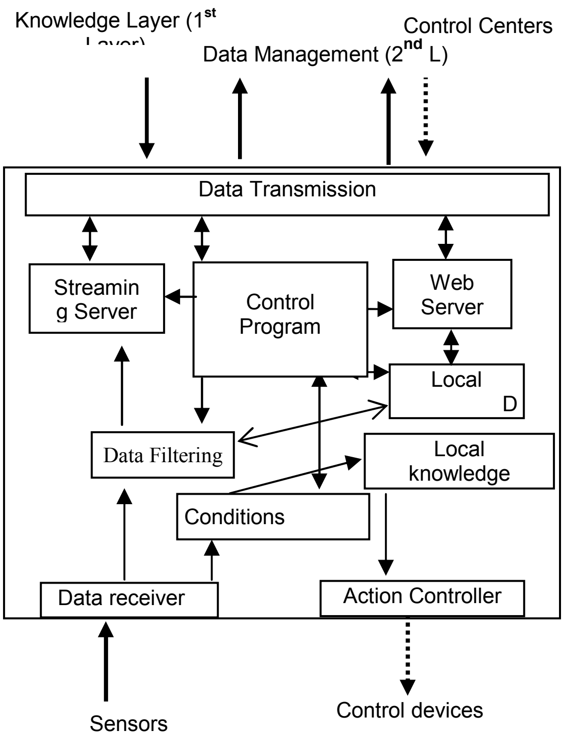

Figure 2. Architecture of the front-end component

4.1 Front-end component objects

The architecture is designed to achieve all of the missions of the component in the 3LOM architecture. There are 10 objects in the component. The descriptions of each object of the component are described in the following section.

The front-end component consists of 10 objects. The architecture shows how the relations and coordination among all of those objects are designed. In addition, this architecture also shows data relations and coordination of this component and middle layer and top layer in 3LOM and control centres.

4.1.1 Data transmitter

Data transmitter component plays roles of transmitting the data to the data management system in the middle layer or any connected control centres. This component also plays important function to receive data from the higher layers and related control centres as well. This component should use TCP/IP standard communication protocol for data communication and can be connected using wire and wireless communication technology.

4.1.2 Data receiver

Various data represent the real condition in the environment generated by appropriate attached sensor. The data receiver in this layer is a component to handle all those attached sensors. Since many types of sensors were developed by various manufactures, this component should use technology that has capability to accept various data and signals from various sensor devices of different manufacturers.

4.1.3 Data filter

The filtering unit receives raw data from each connected sensors from data receiver unit. There are instances where the data acquired needs to be transformed in a more meaningful way for use by this system, for example, noise reduction or some statistical processes. Filtering processes are done according to the need of the systems. Some data are used by this front-end and some others data used by middle layer or related control centres.

4.1.4 Local data manager

Data from sensors usually arrives into the local controller non-linearly and continuously. We need to manage this data and be able to use it for special purposes. Some processes such as condition tracking of specified sensor, accuracy checking or knowledge generation as well as routine data requirements need history of data generated by sensors. As an autonomous system, the front-end component needs to manage its local data and use its own database system. In general, this unit has the role of local data management produced by other components in this unit.

4.1.5 Local knowledge manager

Local knowledge manager is a component to manage local knowledge and customized those ones for some special missions related to the real condition of the specified location. Some basic operation like how to add, update or even delete some knowledge is provided through this component. Global knowledge generated by top layer will be distributed to all of the front-enc components. This component also manages the process of accepting new global knowledge from the knowledge management layer in the top layer.

4.1.6 Control Program

The control program has responsibilities to handle all objects inside the front-end component. Bu using this control program, we can start all services within the component. Using this unit also allow us to customize the component according to specific role of each front-end component plays.

To provide access to this unit from remote location, customization scripts are also defined. These scripts can be used using both direct remote connection and embedded web server installed in this unit. The purpose of these scripts are not only related to the customization processes but also can be extended to some execution tasks sent by central control system for special purpose.

4.1.7 Conditions detector

This component plays an important role in handling and identifying conditions from specific sensors connected to the data receiver component and sends appropriate data to the local knowledge manager. Then the local knowledge data manager will use this data and knowledge to generate some actions. Condition detector in this component is a specific artificial intelligence designed to detect condition, ensure the correct definition of environment condition and send the report to knowledge manager.

4.1.8 Action controller

If a condition has been detected by condition detector, knowledge manager will determine appropriate actions of that condition. Actions can be single action or sequential actions, synchronous or asynchronous actions. Action controller this component performs special tasks to execute tasks related to the conditions which are detected. An example may be the function of a local fire detector unit which may produce an audible alarm followed by information in the electric board about the fire

4.1.9 Web server

Nowadays web server is almost a standard of many devices, for examples many routers or network camera can be set up and customize using a web. By using service by web give us greater possibilities in accessing the device. In this case, putting a web server inside this device is very important step to give advance facilities to handle the device.

Web server is created to give another access to the local unit of each location using web instead of socket connection. We can use the web server to provide some controlling services of the local environment, run customization scripts or browse the current conditions in specified location. Nowadays web server support not only web page information but also other service such as web service as new technology for distributed system.

4.1.10 Streaming Server

The role of the front-end components for data stream management systems is to provide streaming data for simultaneous request through the internet. Each of front-end components will play roles of data provider for multi data stream systems, for example sensors data for real-time monitoring by fire office.

For this approach, using socket communication is backbone of any streaming system through the internet. To be more detail, each of front-end components will provide multi data stream systems which can be accessed from remote location. Each front-end component should capable to handle multiple connections from any request. However user identification processes should be performed before provide the service.

4.2 Front-end component design modelling

In order to define clearly all roles of the component, we need to create model of all processes related the responsibilities of the component inside the 3LOM architecture. The model shows how all processes in the component are defined and works.

4.2.1 Use-case diagram of the component

Sensor and actuator actors represent devices managed by the component. Sensor devices interact with system through the data acquisition process while the actuator will get command from the generate action process. The data filtering process will perform filtering process for eliminating noise and forming formatted data. Furthermore, both sensor and actuator can be customized by the customization process.

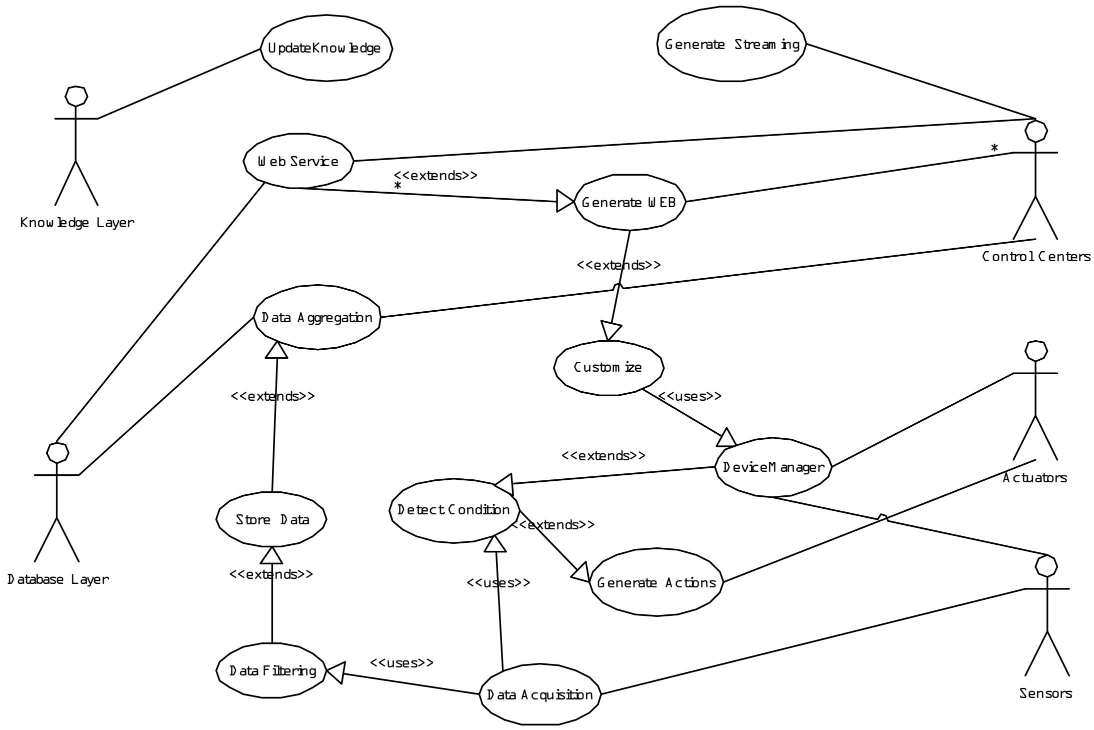

Figure 3. Use-case diagram of the front-end component

The component also performs monitoring to all sensors and actuators to check status or any changes on the devices done by the device manager. On the other hand, the control centres can use the data provide by the data manager in the component and perform their monitoring task to the device.

We also use the web service to develop the web system of the component where control centres or other user can access the component to monitor device or customize the devices. Moreover, the component also create streaming server to provide streaming data service and handle real-time requests from the control centres.

5 Front-end component integration and synchronization design

In developing such a huge distributed system, integration strategies are important steps to realize such multi-missions systems. In the system design, standard for data integration including access mechanisms will enable each organization to develop their own systems to implement the integrated system.

According to those goals, we need to consider design processes to integrate the front-end component with other layers in 3LOM architecture and also other front end components. In this section we will discuss more detail about integration aspects of this layer and related information of the integration aspects including communication system, data synchronization and access mechanism.

5.1 Front-end component integration

The main role of the front-end component within 3LOM architecture is as an interface to the environment, acquiring data and send controlling command; we need to describe also synchronizations of this component with other layers and other components.

There are 4 main components in the system which interacts with the front-end component. The front-end component has to acquire data and distribute information to the environment. It also communicates with other front-end components and exchange sensor data and signal status or can be extended to more high level information. The front-end component also sends high level information as statistical data to the database layer as well as traffic data to the control centres. Moreover it also receives updated knowledge from the top layer

For the purposes of integration of front-end components and other layers and other front-end components we describe data integration of our approach as follows

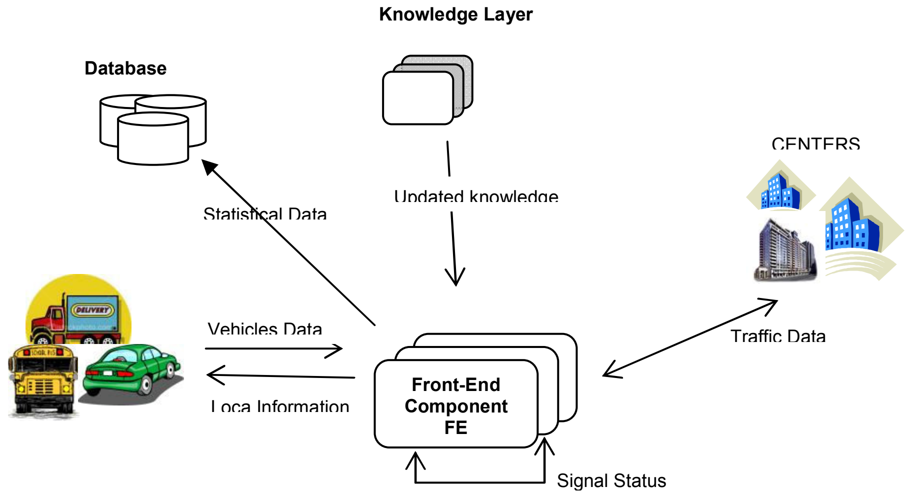

Figure 4. Design diagram of data synchronization

5.2 Front-end component access mechanism

As nowadays the internet technology has grown and developed significantly, we want to use it as backbone of the integrated system, furthermore accessing protocol of this component also based on that technology.

Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks. Web service is a software system identified by a URI (RFC 2396), whose public interfaces and bindings are defined and described using XML (Zachary Et All, 2002).

As our design use internet technology, real-time data will be provided by the front-end component by using internet data streaming technology. The role of the front-end components for data stream management systems is to provide streaming data for simultaneous request through the internet. To be more detail, each of front-end components will play roles of data provider for multi data stream systems, for example sensors data for real-time monitoring by fire office. For this approach, using socket communication is backbone of any streaming system through the internet.

In addition, embedded web server gives accesses through internet to other user of transportation system such us pedestrian, driver, traveller etc who need information related the condition in localized environment of a front-end component. This web service can be extended to give additional service through wireless system and provide connection access trough other smart device such as mobile phone or PDA handset or even in-vehicle navigation systems or common monitoring from related control centres.

6 Application prototype

In this paper, we try to create a prototype and implement of front-end components for the integrated transportation systems. We implement this prototype in order to implement and evaluate this model for better improvement in the future.

As a prototype, we have not implemented all of requirements we described in the previous section before. We begin with using open standard architecture to handle device and controller using LONWORKS platform and LONWORK devices (Echelon Corp, 2001) to simulate sensor and actuator management. We also implement prototype systems to access the front-end component. We use 3 ways to communicate and access this component, using socket communication, internal web server and SOAP for web XML web services.

6.1 Streaming service using TCP/IP socket communication

The basic data structure for devices is quite simple since the main purpose is to provide recent data of devices within the front-end component. To be more detail, each of front-end components will play roles of data provider for multi data stream systems, for example sensors data for real-time monitoring by fire office, C02 level and etc.

All devices data of the front-end component are stored in the local database. Whenever the front-end component needs to start the system and provide streaming accesses, it will read devices data in local database including number of devices and create data structures in the memory according to the number of devices inside the front end component.

For this prototype we have made a simulation to provide a streaming of temperature sensor from a LonPoint device inside the front-end component. Front-end component create a tcp listener to handle multiple request connection from client. We show to simulate this streaming by using 4 clients as shown in the following figures while 2 other clients are located in different computer

The picture shows the simulation of how this front end component distributed data to multiple users at the same time. Clients are representation of user who needs to get streaming data of temperature sensor data from this component. For those clients we also create a simulation how we can use the streaming data to provide a real-time graphic of sensor data.

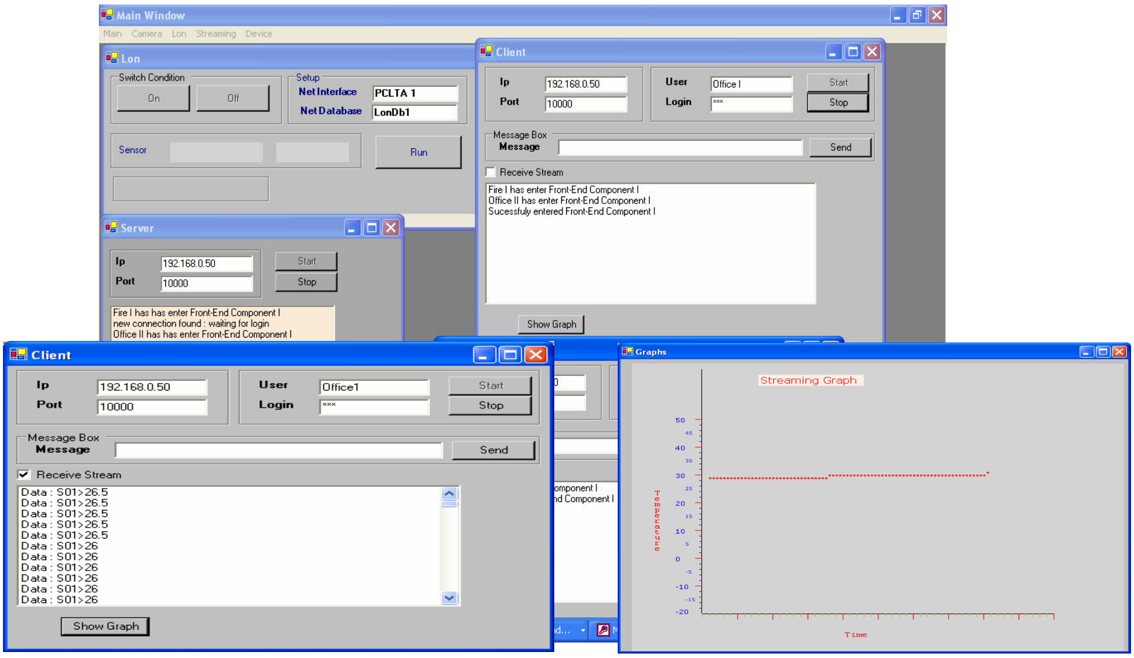

Figure 6. Streaming simulation for a sensor

The figures in the bottom show an example of a streaming client and the related streaming graph. For this prototype to provide streaming data of temperature sensor, the server inside the front-end component will deliver string data with format “Data : deviceName > value “ , for example “Data : S01 > 26.5 “ is shown that the streaming data is from device which has S01 name with the value 26.5 . Based on that simple format, each streaming client can generate appropriate graph for each device as shown in the figure above.

In this prototype we also provide access through a web system, using web systems has some advantages which make easier to access the component by using internet browser only. By providing this access we have advantages of easy and flexible access from any independent platform or various types of devices. We created the web page using data provided by the web services we have created as a part of services of the front-end component.

6.2 Web Access

In this prototype we also provide access through a web system, using web systems has some advantages which make easier to access the component by using internet browser only. By providing this access we have advantages of easy and flexible access from any independent platform or various types of devices. We created the web page using data provided by the web services we have created as a part of services of the front-end component.

Using web access approach gives us some possibilities to extend services of the front-end component to the external users which access the systems through outside of ITS network such us driver or traveller. In this prototype, we separate access privileges based on login type; web system read the database to check the access type of the user and provide appropriate information of the component. An access to the component can not be done without passing login authorization

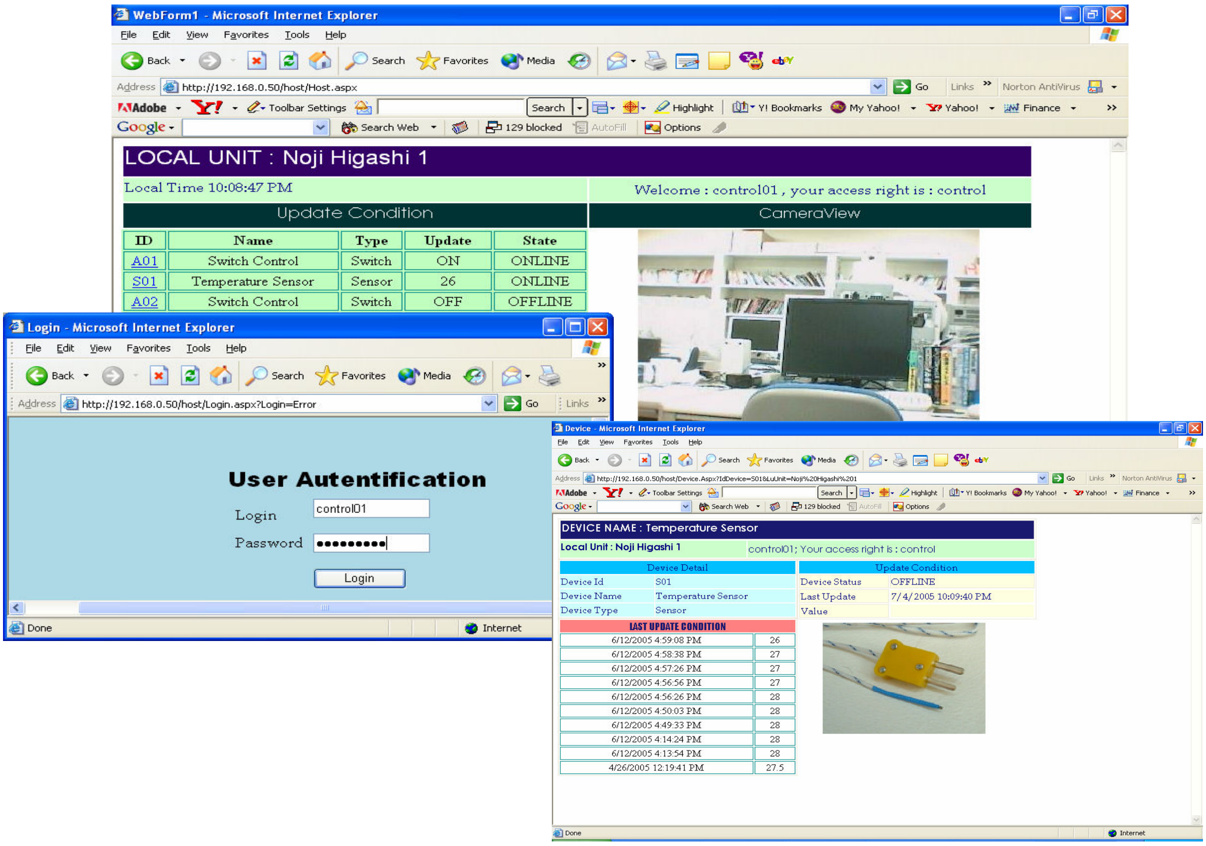

Figure 7. Web access to the component

7 Evaluation

This section shows evaluations of front-end component and 3LOM architecture. We analyse the performances by evaluating bandwidth consumption, response time, flexibility and interoperability in development and robustness of the system.

7.1 Bandwidth consumption and response time

Most of recent transportation control systems nowadays are centralized systems. In centralizes systems, control right and monitoring is put into control centres. Most of those systems, control centres decided decisions and send it directly to each device.

The process of data acquisitions are done in device level and processes related to that ones are done in control centres. Then control centres do processing of those data and distributed decisions to the entire systems.

SCATS is one of popular traffic control systems nowadays. SCATS gathers data on traffic flows in real-time at each intersection. This data is fed via the traffic control signal box to a central computer. The central computer makes incremental adjustments to traffic light timings based on minute by minute changes in traffic flow at each intersection (M. Dineen, V. Cahill 2001). Using the systems above, all data need to be transferred to the central computer, subsequently the central computer needs to create decisions and distribute those to each control device back. As a result, it will increase network bandwidth consumption use to send data from all sensors in city and distributed information back again to each related actuators.

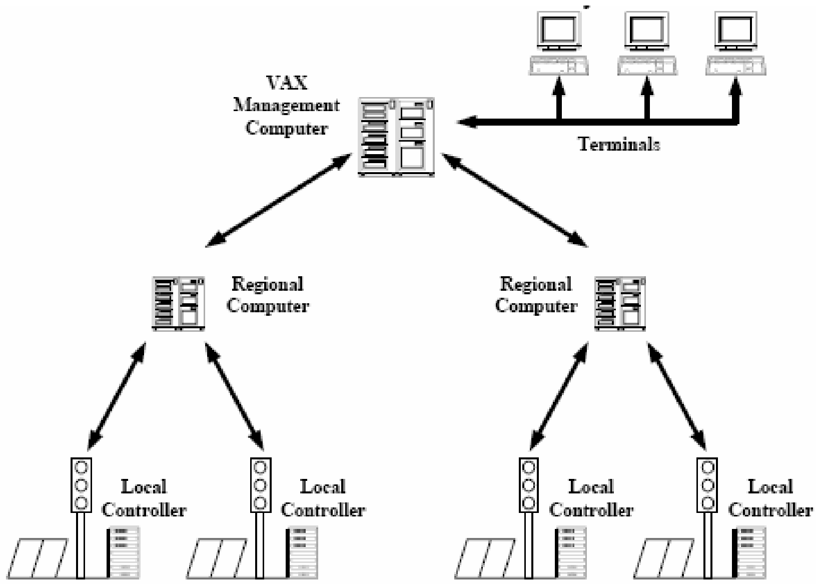

Figure 8. SCATS physical architecture

We proposed 3LOM architecture as distributed system architecture.. As a bottom layer in the architecture, front-end components are autonomous distributed components. We have described in previous section that each front-end component has local knowledge which works and coordinate with local detection to provide appropriate actions and establish an autonomous system. The following figure show how the front-end component works in the 3LOM architecture.

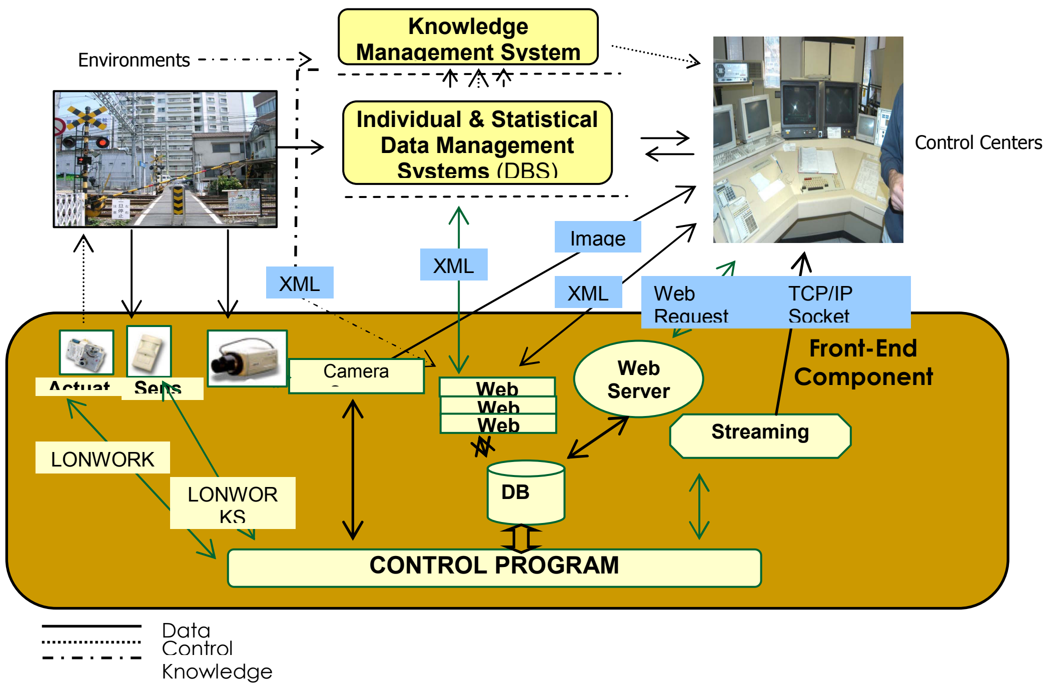

Figure 8. Prototype model of system integration

Based on those approaches, each front-end component will not send all sensors and actuators raw transaction data to central computers like in centralized systems. Each front-end component will play roles as streaming server to any control centre that need streaming data for their purpose. In addition, front-end components are capable to provide device data based or request given by any control centres as well as statistical data of each front-end component will be sent to the middle layer or control centres periodically.

For real-time systems, using decentralized systems rather than centralized ones will increase the response time for generating responds to localized environment where each distributed component located. Each front-end component will generate decisions using installed local knowledge which can be updated by knowledge management layer.

Control centres can send instruction data to each device of the component but the local system inside the component will decide and perform execution for the task. Responds to localized environment will be generated immediately while local knowledge in coordination with condition detector recognized defined condition in localized environment and find best responses to the environment.

Using these approaches, the responses time of integrated systems to transportation systems which are very dynamic and high frequency of changing data will be faster comparing with the centralized systems.

7.2 Robustness

Since transportation systems are one of most fundamental systems in the society, robustness aspect of such important system like transportation system is an indispensable need. There are many conditions which will affect to performance of system, for example such as communication problem, sensor damage, broken actuator and any specific problems which affect directly to the performance of the system etc.

For those missions, we designed 3LOM architecture as distributed systems architecture which enable us to distribute control right and responsibility related to transportation environment condition. Moreover, each front-end component in the system has local knowledge which enable each front-end to work with its minimum function whenever there were problems occurred. The local knowledge of each front-end component can be customized according to role of the component in the environment.

Furthermore, we need to design general local knowledge that can be customized and adaptively changed according to the problems occurred. Based on its customized knowledge each front-end component will work in the integrated system and perform its services and control according to the environment or mission the component have.

7.3 Flexibility and interoperability in development

For such integrated systems, the issues of flexibility with high interoperability must be placed as one of important guidance in system development. In integrated transportation that will involving many different organizations, various separated systems high flexibility in system development will minimize development time and cost of integrated systems and reduce complexities due to limitation of specs or platform technology. The flexibility can be achieved if we provide high interoperability among systems and device of the integrated systems.

There are many incompatibilities and difficulties to interconnect digital controllers from different manufacturers since there are no standard protocols; as a result every system must have custom application programs and results low interoperability. For such huge integrated systems we need technologies which overcome those difficulties and provide flexibilities of device problems as interfaces to the environment. We consider technologies which implement open standard systems which enable to all of the manufactures to refer the specification and create devices which provide high interoperability among many devices from different manufacturers.

One of difficulties to build such an integrated system, most organizations still uses their own systems for managing data and information. From the view of technology, complexity and scale of these systems have made the difficulty to implement integrated system idea. Another important factors are most of those systems were start to be developed when technology for data communication and integration not as advance as nowadays.

In the prototype we design 3 standard accesses the component by using web access, socket communication or XML web service. By providing those three standard accesses, each client office who wants to develop application systems can flexibly design or integrate their systems with this component as we have shown in figure 8.

8 Conclusion

In this paper we have proposed an architecture design of the front-end component for the integrated transportation system as the bottom layer within the three layers object model for integrated transportation systems. The front-end component plays roles to acquire information and distribute decision or information to its controlled environment which makes the indispensable roles of the component within the system.

We have designed the front-end component as a standard and customizable multipurpose device for the integrated system. This approach has been chosen to achieve high flexibility and feasibility in development of such a huge and complex system such as integrated transportation system. Moreover, using standard device for the entire system will increase high interoperability and reduce cost for the implementation.

In addition, since the integrated transportation system is a complex and huge system, we propose that the front-end component should provide open standard access mechanisms to enable flexible used and high interoperability of the component for further integrated system development.

To accomplish the complete front-component system, for the future research work, we consider some researches topics need to be explored:

1. Implementation of the front-end component as an outdoor device which enable the device to be used in transportation systems.
2. Research and development for real-time traffic simulation and establish a data stream management system for transportation system as an important breakthrough for the integrated transportation system modeling.
3. Research and development of the knowledge layer of the integrated system which enable the front-end component to implement it local knowledge as a part oh the knowledge layer as the global layer

We recommend that those three research topics recommendation are important topics or the implementation the complete front-end component and the integrated system as the main mission of the project.

REFERENCES

American Association of State Highway and Transportation Officials (AASHTO), Institute of Transportation Engineers (ITE), National Electrical Manufacturers Association (NEMA) (2002). *NITCP (National Transportation Communications for ITS Protocol) Guide updated version 3*

B. Babcock, S.Babu, M.Datar, R.Motwani, J. Widom (2002) *Models and Issues in Data Stream Systems*. In Proceedings of 21st ACM Symposium on Principles of Database Systems

Bob McQueen, Judy McQueen (1989) *Intelligent Transportation System Architecture*, Arctech House Inc., 1999

Echelon Corporation *Lonworks Open System Design Guide, Designing Open Building Control Systems Based on LONWORKS® Technology version 2.0*, http://osa.echelon.com/Program/PDFs/guidelines/OpenSysDesignGuide.pdf.

H. Ikeda, N. Vogiatzis, W. Wibisono, B. Mojarrabi, and J. Woolley (2004) *Three Layer Object Model for Integrated Transportation System.* 1st International Workshop on Object Systems and Systems Architecture, Victor Habor, Australia.

Ireland On-line, *Adaptive road traffic control systems in use in Ireland*. http://www.iol.ie/~discover/traffic.htm.

M. Dineen, V. Cahill (2001) “Towards An Open Architecture for Real-Time Traffic Information Management”. Proceedings of the 8th World Congress on Intelligent Transport Systems, Sydney, Australia, 2001.

N. Vogiatzis, H. Ikeda, J. Woolley, and Y. He. "Integrated Multi-Nodal Traffic Network Systems". East Asia Society for Transportation Studies Journal (EAST), Fukuoka, Japan, 2003.

N.Vogiatzis, H.Ikeda, W.Wibisono."On the locality scope model for improving the performance of transportation management systems".Journal of Road & Transport Research, Vol 14, No 2, June 2005.

P. Haya, X. Alaman , G. Montoro, “A comparative study of ubiquitous computing”. The European Online Magazine for the IT Professional, Vol. II, No. 5, 2001.

Reza S. Raji. “End-to-End Solutions with LONWORKS Control Technology”. http://www.lonmark.org/solution/home/end2end.pdf .

The World Wide Web Consortium (W3C), “Extensible Markup Language (XML)” . http://www.w3.org/XML/

W. Wibisono, H.Ikeda, Nikolaos Vogiatzis. “Design of Multipurpose Front-End Device for Integrated Transportation Systems”. National Seminar of Computer Science and Information Technology (SNIKTI), Bogor, Indonesia, 2004.

Zachary G. Ives, Alon Y. Halevy, Daniel S. Weld. “An XML Query Engine for Network-Bound Data”. The International Journal on Very Large Data Bases (VLDB), Volume 11, Number 4, 2002.