

Integrated Modeling and Simulation Framework for Freight Transportation in Metropolitan Areas

**Final Report
METRANS Project 08-04**

September 2010

Viktor Prasanna, Principal Investigator

Amol Bakshi, Principal Investigator

Ming Hsieh Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089-2562



Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, and California Department of Transportation in the interest of information exchange. The U.S. Government and California Department of Transportation assume no liability for the contents or use thereof. The contents do not necessarily reflect the official views or policies of the State of California or the Department of Transportation. This report does not constitute a standard, specification, or regulation.

Abstract

Freight transportation at distribution nodes such as marine ports, airports and rail yards contributes to congestion, environmental pollution, traffic accidents and has been putting tremendous pressure on metropolitan areas to deal with such issues. A prerequisite for proposing any solution that would make the existing systems more efficient is an accurate analysis and understanding of freight movements. Economic forecasts are unanimous in predicting the need for additional capacity in order to keep pace with rising demand. Adding capacity by following the traditional way of building new infrastructure is no longer as feasible as it used to be especially in Metropolitan areas where the scarcity and cost of land are obstacles difficult to overcome. A number of modeling and simulation tools have been developed for studying specific traffic systems. While a single model cannot capture all aspects of freight transportation which interact and affect each other in a complex manner. For example, the freight movement nearby a marine terminal cannot be captured solely by the terminal simulation or nearby roadway network simulation. Integration of legacy simulation and analysis tools along with holistic optimization is thus necessary to study complex transportation scenarios. Such integration also significantly increases the value of individual tools.

The intent of the project is to develop an extensible unifying modeling and simulation framework into which more focused software tools can be integrated to address complex transportation problems. We adopted model-based integration as the core design principle and semantic web as the integration technology. A domain ontology model was developed as the common modeling language of simulation scenarios and as the shared data model for application integration. It provides a single logical space for simulation design while hiding the disparity of underlying simulation tools. The framework is designed to be modular and extensible. Researchers developing simulation models, control algorithms etc., can insert new modules into the overall environment with little effort and synergistically with other existing modules. We demonstrated our system by integrating existing models developed by METRANS researchers such as: (i) the terminal and traffic network models by Prof. Ioannou and Dr. Chassiakos, (ii) the highway ramp control model by Dr. Hwang and Prof. Ioannou. Especially, we interacted with other METRANS researchers as well as engineers in the field in an effort to establish our simulation environment as the core of a test bed for studying freight transportation in the Los Angeles metropolitan area.

Table of Contents

1. Introduction.....	1
2. Background and Challenges	4
3. Use Cases.....	6
4. Integration Framework	8
4.1. Overview of the Framework Architecture	8
4.2. Ontology-based Modeling Integration	10
4.3. Workflow-based Application Integration	12
5. Empirical Evaluation	14
5.1. Marine Terminal – Traffic Network Simulation.....	14
5.2. Highway Ramp Control Simulation.....	17
6. Conclusion	19
7. Implementation	20
8. References	21

List of Figures and Tables

FIGURE 4-1: ARCHITECTURE OF THE INTEGRATION FRAMEWORK.....	8
FIGURE 4-2: DOMAIN ONTOLOGY.....	9
FIGURE 4-3: SCENARIO EDITOR	11
FIGURE 4-4: SIMULATION WORKFLOW PARADIGM	13
FIGURE 5-1: MARINE TERMINAL – TRAFFIC NETWORK FREIGHT TRANSPORTATION SCENARIO	15
FIGURE 5-2: CONTROL LOGIC OF THE INTEGRATED TERSIM-VISSIM SIMULATION.....	16
FIGURE 5-3: LAYOUT OF THE FREEWAY MODEL	17
FIGURE 5-4: RAMP QUEUE LENGTH WITH (A) NO CONTROL (B) RAMP CONTROL	18
TABLE 5-1: RAMP CONTROL SIMULATION RESULTS.	17

Disclosure

Project was funded in entirety under this contract to California Department of Transportation.

1. Introduction

Globalization, international trade and economic development have led to an increase in the volume of traffic flows by all transport modes. Traffic at ports, rail yards, warehouse distribution nodes and metropolitan areas contributes to congestion, traffic accidents and environmental pollution and puts tremendous pressure to deal with such issues. A prerequisite for proposing any solution that would make the existing transportation systems more efficient - especially in complex environments such as in a metropolitan area - is a careful and accurate understanding and analysis of traffic flow. Economic forecasts are unanimous in predicting the need for additional transportation capacity or control strategies in order to keep pace with rising demand. Adding capacity or introducing technologies by following the traditional way of building new infrastructures is no longer as feasible as it used to be especially in metropolitan areas where the evaluation and implementation cost are obstacles hard to overcome.

The state-of-the-art method for studying transportation systems is simulation modeling which affords the opportunity to evaluate control and design strategies without committing expensive and time-consuming resources to implement alternate strategies in the field. Major efforts have been invested in developing traffic simulation systems during the past a few decades. Most of them are based on a mathematical approach which consists of reproducing vehicle streams in specific areas at a certain granularity [1]. For example, terminal simulators model special infrastructure inside a container terminal – ports, yards, gates, logic of freight flow, etc. – in an attempt to simulate traffic flows inside terminals accurately. These include marine port simulator TerSim [2] and air port simulators [3]. Simulators that model traffic on roadways can be classified as macroscopic, mesoscopic or microscopic. Macroscopic models [4,5] tend to model traffic as a continuous flow, often using formulations based on hydrodynamic flow theories. Mesoscopic models [6,7] model individual vehicles, but at an aggregate level, usually by speed-density relationships and queuing theory approaches. Microscopic models [8,9] capture the behavior of individual vehicles and drivers in great detail, including interactions among vehicles, lane changing, and behavior at merge points. Increasing the modeling detail and hence the granularity of simulation improves accuracy of the simulation result but also increases the complexity of computation and sensitivity to modeling errors.

A holistic evaluation of the impact of different transportation policies in a complex scenario such as the freight transportation in metropolitan areas requires a comprehensive simulation environment which models all aspects of the target system. This by itself is a very challenging problem due to the presence of multiple modes of transportation which

interact and affect each other in a complex manner and cannot be captured by a single existing simulation model. To facilitate the introduction of new technologies and methods to improve transportation efficiency and keep pace with rising demand, a framework for integrated traffic modeling and simulation which supports integration of a multitude of simulation tools is needed.

Advances in information technologies, data management and computation tools, together with recent research in systems optimization and control, makes possible new approaches to solve the problem. In this project, we developed *a semantic framework for integrated traffic modeling and simulation*. In this project we refer integrative as the integration of software applications, which is a common research area in software engineering. Traffic simulation tools were originally designed to simulate specific aspects of traffic systems such as a marine port and a traffic network independently. Our work focuses on building a platform into which heterogeneous traffic simulators and computation tools can be integrated and used to study complex scenarios such as freight transportation in metropolitan areas. Key problems addressed by the developed system are:

Model integration: Multiple simulation models capture different aspects of a transportation system. One key goal of our work is to enable the evaluation of complex simulation scenarios where each scenario represents a certain realization of the system or a certain operational strategy. An evaluation of the scenario is a forecast of the system response and an optimization chooses the ideal designs or operations. Analysis at the system level necessitates that the data in the many models and the results of their simulation be merged in meaningful ways.

Model management: A traffic system could have alternate models depending on the simulators and modeling approaches used. An alternate model may also represent a variation of the system with a different set of values of parameters. The integration framework should provide services to manage legacy models including search, view and retrieval for reuse.

Application integration: Researchers and engineers may use different simulation tools to model traffic systems, and use a variety of computation tools to develop control algorithms and data analysis models etc. The integration framework were designed with modularity and extensibility in mind. Domain experts are able to insert heterogeneous tools into the simulation environment to model a hybrid system with little efforts and without affecting other modules. Therefore the integration system is an evolving framework allowing additions, improvements and changes on a modular basis while maintaining the integrated nature.

User focus: It is also important to note that the ultimate purpose of the integration framework is to assist domain scientists (e.g., control theorists) who may be not computer experts in the rapid prototyping and execution of integrated simulations. Our experience

has shown that “soft” factors such as the design of user interfaces, novel visualization tools, intuitive workflow wizards, etc are as important from the end users’ perspective as the underlying technological solutions.

The problems addressed by our framework touch upon some of the core issues of information integration, i.e. *providing an integrated and coherent view of data stored in multiple, possibly inhomogeneous information sources* [10]. Our system aims to provide a unified environment to define and evaluate traffic simulation scenarios. Each integrated simulation model maps to multiple application-specific models which can be executed by corresponding simulation tools. A domain ontology acts as the key component to enable model and application integration, and forms the basis for the intuitive user interfaces that allows transportation researchers and engineers to refine their solutions.

2. Background and Challenges

Integrated use of traffic simulation tools and other software applications are among the new trends in solving transportation problems, e.g. evaluating the impact of terminal handling strategies on local traffic conditions and terminal throughput [2]. Existing integration systems can be categorized as homogeneous or heterogeneous based on whether they integrate simulators of the same type (e.g., traffic network simulators), or they integrate simulators modeling heterogeneous component systems (e.g., traffic network simulator and marine terminal simulator). A few examples of homogeneous integration are MEZZO-VISSIM [11] and Paramics DYNASMART [12]. The VISSIM-TerSim [2] test-bed and TraNs [13] are an examples of heterogeneous integration system.

MEZZO-VISSIM simulates different areas of a traffic network at different levels of detail. It combines the strengths of meso simulation of large scale networks with less sensitivity to modeling errors and less calibration with those of micro simulation of interested areas in greater detail. The integration architecture consists of a new module outside the meso and micro modules that contains some components shared by all the tools: a database with information of the entire network graph, the travel behavior component with route choice models and path generation algorithms. Each time a simulated vehicle makes a route choice, the common module is consulted. Both simulators also update the network database regularly with link conditions in their sub-network.

Paramics-DYNASMART combines two simulators in an embedded structure. The primary objective of the integration is to provide a better route-choice capability for the micro simulation of Paramics by using the path dynamics of DYNASMART. The framework consists of several additional modules such as data communication and routing decision modules to transmit vehicle route decisions processed at the abstract network to the detailed network.

VISSIM-TerSim test bed allows joint modeling and simulation of traffic in marine terminals and roadway networks. It consists of three modules: TermSim, VISSIM and TermCost. An external program is used to execute VISSIM COM commands to access its simulation data, allowing VISSIM to work as an automation server to export objects. TerSim employs a client program to collect data from VISSIM through the server interface and converts it into its inputs. Also TerSim writes its outputs as the corresponding input object of VISSIM. The TermCost is an offline cost model to analyze the simulation data.

TraNS, as a heterogeneous integration platform, combines the capabilities supported by a network simulator ns-2 and those supported by a traffic simulator SUMMO for realistic evaluations of VANETs (vehicular Ad-Hoc networks) applications. TraNS employs the Traffic Control Interface (TraCI) module for interlinking road traffic and network simulators. TraCi module translates the information exchange within VANET simulated by ns-2 into atomic mobility commands and feeds the signal to SUMMO to manipulate the mobility of individual vehicles.

The key difference between our effort and the ones mentioned above is that our framework is not targeted for a single problem and a single class of end users. In the above systems, a fixed set of tools are tightly coupled through data exchange modules and reconfiguration of the system for different simulation workflow needs extensive reimplementations. The framework for integrated transportation modeling and simulation developed in this project offers benefits of modularity, extensibility and reusability. Researchers developing different simulation tools, optimization and control algorithms, data analysis models, etc., would be able to insert components into the overall simulation environment with little effort and without affecting other modules. Therefore the system is an evolving framework allowing additions, improvements, changes on a modular basis while maintaining the integrated nature. As end users, domain experts can define and simulate a variety of operational scenarios, and view the associated forecasts as an aid to decision making.

There are many challenges in realizing the above framework vision, some technical and some organizational. One of the key technical challenges is building a loosely coupled architecture which allows independent simulators, databases, real-time measurement and business components to communicate with each other during simulation time. Complex issues can arise such as data heterogeneity, tool interoperability, etc. Inspired by a wealth of work on model/ontology based information and tool integration [14,15], we have adopted semantic web technologies to design the integrated simulation system. Modeling schemas are used to capture the freight transportation domain concepts in an ontological framework. Target applications are modeled by workflows of integrated tools. The ontology describes the semantics of the high level workflows and also provides a query model for legacy tools to communicate. To the best of our knowledge, our work is the first to use model/ontology technique to address integration problems for transportation systems modeling and simulation.

3. Use Cases

Our framework allows a transportation domain expert (e.g., a researcher, a city planner or port supervisor), who may not be familiar with the implementation details of the entire set of simulation tools and who may not necessarily be a software integration expert, to integrate simulators, design simulation scenarios and evaluate system-wide impact of local changes. In this section, we discuss illustrative use cases that justify our integrated modeling and simulation framework for freight transportation systems. In addition to highlighting the utility of the framework, the types of use cases we address also influence the software architecture design and user experience. In our example, we used model data from the Los Angeles metropolitan area that includes the Long Beach port and the roadway network that serves freight traffic to and from the port.

A number of relevant use cases can be defined in this setting. For instance, the port management could be interested in understanding how the port operations and the nearby roadway network will be affected if key pieces of infrastructure (e.g., loading cranes) cease to operate temporarily. In addition to understanding the impact of breakdown, there are other use cases that relate to proactive planning of port capacity and operational strategy to handle projected increase in the amount of freight handled by the port. For example, city planners will need to understand potential bottlenecks in the system as freight traffic continues to grow – will an increase in the amount of cargo moved by rail reduce the roadway congestion in the areas surrounding the port? Where should inland container storage facilities be located to maximize truck utilization and reduce the average time a truck has to wait in traffic and within the terminal? How many additional incoming shipments can the port handle before the terminal infrastructure becomes a limiting factor? If the freight transportation supply chain has to be de-bottlenecked, which section of the infrastructure should be overhauled first?

Clearly, no single model or simulator can capture the entire freight transportation system – e.g. a traffic network simulator cannot simulate the traffic flow inside a terminal while a terminal simulator cannot capture the freight movement outside the terminal. Also, it is unreasonable to expect the end user to be proficient in the domain and also be an expert in using the entire set of simulators that need to be configured and executed in concert to answer questions such as those listed above. To allow users to rapidly and intuitively specify and evaluate the above scenarios, we focus on defining application-independent abstractions that capture the intent of a use case in terms of input and output. The scenarios specified through these abstractions are then translated into specific configurations and commands to integrated simulators.

The input data set for a typical simulation can be divided into two main categories: model information and control events. For the examples, model information may consist of parameters describing the attributes of terminal infrastructures, road networks, etc. Control events like the “crane failure” need to be mapped to the corresponding set of time/model parameter settings. The simulation outputs for a given scenario are values of evaluation functions of modeling variables, which could be the port throughput or the average traffic flow rate near the terminal. Graphs can be plotted based on the output data as per user’s requirements. The configuration of our integrated modeling and simulation framework can therefore be divided into two steps: preparing model inputs, and defining control events and outputs.

Prepare model inputs: Users provide model information of a complex target system by composing a set of base models. For the crane failure use case, users can choose TerSim to model Long Beach port and VISSIM to model the Los Angeles roadway network. The base set of models for the port and the roadway network are ‘published’ into our framework to form a *model inventory*. Users can select the models from the *model inventory*, enabling the reuse of base models for composing a variety of system-wide models.

Define control events and outputs: End users configure a set of “what-if” simulation scenarios to explore the relationships between control events and simulation outputs. Each scenario is bound with a simulation workflow or target application model that is composed using the model inventory. For the crane failure use case, users should be able to create a scenario control labeled “crane failure” and associate it with a set of time/model configuration information for the TerSim simulator. E.g., 100% capacity could indicate normal operation till, say, 10am, 30% capacity from 10am-5pm indicates crane failure, 150% capacity from 5pm-7pm indicates repair of the failed cranes and also extra capacity made available to handle the backlog. Users are able to create various such scenarios at high level to examine the input-output relationship and aid rapid decision making.

A significant advantage of the above two step development flow is the reduction in cost of application definition. If each scenario has to be constructed from scratch, the number of scenarios that can be defined and analyzed in a given time becomes significantly longer compared to our approach where a bulk of the definition and source data already exists in the system. The inventory also acts as a single version of the “truth” and prevents the errors caused by data duplication and inconsistent modification.

4. Integration Framework

Our framework for integrated traffic modeling and simulation enables the integration of heterogeneous traffic simulators, generic algorithms and field data into a unified environment. We envision the integration system to be an evolving framework allowing additions, changes of underlying components on a modular basis while maintaining the integrated nature. Some of the key design objectives are:

Single view of information: The framework hides the disparity of data formats, simulation models and tools from end users and provides a single logical space for simulation design.

Generic and reusable architecture: The framework is configurable for a variety of simulation scenarios without the need for extensive redesign or reimplementation. Legacy simulation models can be searched, integrated and re-configured to create new simulations.

Loose coupling of simulation tools: An ontology model serves as a mediator schema which enables loosely coupled tool integration architecture because all the tools can now read from and write to a shared database. Indirect coupling of disparate tools leads to a modular and highly extensible framework.

4.1 Overview of the Framework Architecture

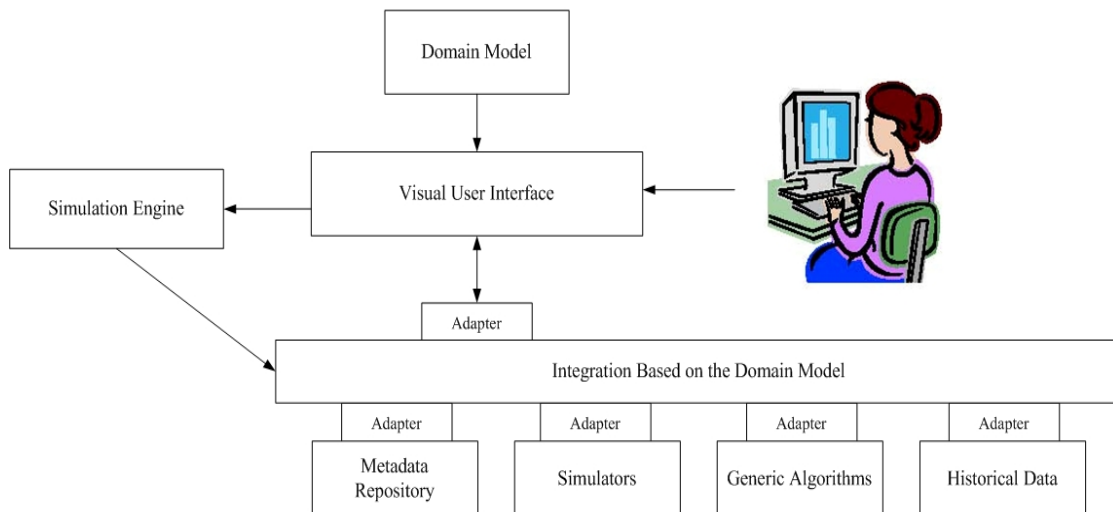


Figure 4-1: Architecture of the integration framework

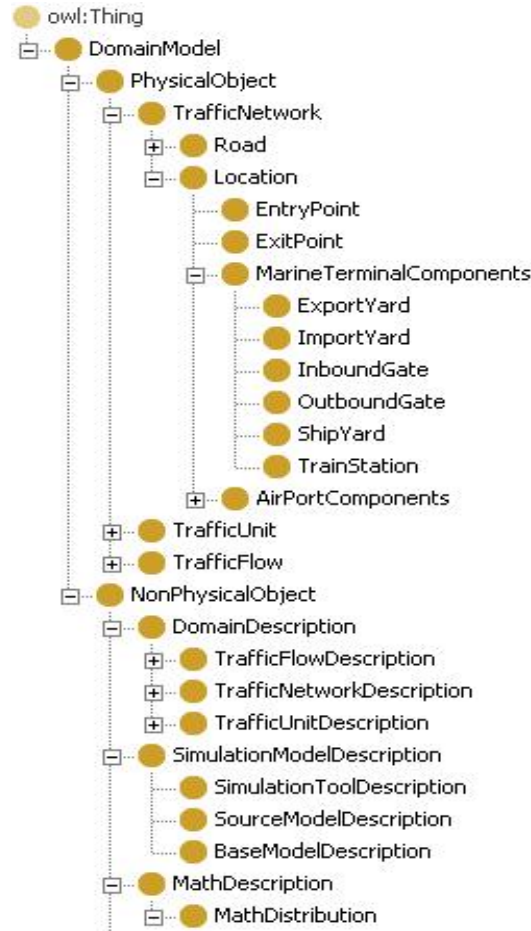


Figure 4-2: Domain ontology

We adopted a model-based approach to achieve the design objectives. In this approach, a semantic domain model is used as a primary enabler of data and application integration. Figure 4-1 shows the architecture of the system [16]. It consists of four major components: the domain ontology model as shown in Figure 4-2 [17], metadata repository, simulation engine and user interface. The domain model captures domain concepts and serves as the mediator data schema for integration of heterogeneous data and models. Data disparity is addressed by mapping between generic data/models and the domain ontology. The metadata repository stores information about simulation data and models in a canonical form that will be programmatically accessible to any component (e.g., simulators, algorithms) that wishes to read from or write to this database. Execution and data flows of integrated simulations are modeled as workflows using pre-defined workflow paradigms. Workflows can be executed by the simulation engine which handles the execution sequence and data exchange of simulation tools.

4.2 Ontology-based Modeling Integration

Traffic simulation tools employ a wide variety of models (e.g., terminal model, link-connector network model and link-node network model) to capture transportation domain aspects. Most of simulation models are stored in weakly structured formats, maintaining (keeping data and model collections consistent, correct, and up-to-date), understanding, searching and integrating model information are difficult problems which have not been well addressed before.

Semantic Web [18] provides an ontology-based framework that allows data to be shared and reused across model, application and even community boundaries. It has been used for data and application integration in various domains such as health care [19,20], biology [21] and oil industry [22]. An ontology is a collection of key concepts and their inter relationships collectively providing an abstract view of an application domain. It defines a common vocabulary across the system and forms the conceptual basis for integration. Ontology-based system design is especially valuable for the design of complex computer-based systems that involve interactions among applications which are not specifically designed to interact with each other. It requires to formalize relevant domain knowledge as a meta-model which represents a set of conceptual building blocks (entities) and the relationships between the entities.

In our framework, using the ontology language OWL [23] we defined a traffic model which captures domain concepts, their attributes and relationships. OWL is attractive for defining domain models because it is built on open standards, provides the requisite level of expressiveness to capture the rich relationships in the transportation domain, has precisely defined semantics and is accessible through flexible APIs. In general, we classify the domain objects into physical and nonphysical classes. Physical class has subclasses like traffic network, terminal components, traffic units etc. Nonphysical objects include property objects such as traffic network description, routing decisions, ship schedule etc. Domain ontology is an evolving paradigm which can be extended by domain experts to capture new domain concepts to meet integration requirements. As shown in Figure 4-2, to allow model a target traffic system involving metropolitan roadway networks and marine terminals, both network components and marine terminal components such as road, ship yard, train station, import/export yard and inbound/outbound gate need to be defined in the domain model. The attributes of domain elements such as length of road, processing rate of inbound gate, capacity of import yard etc are also modeled in the ontology.

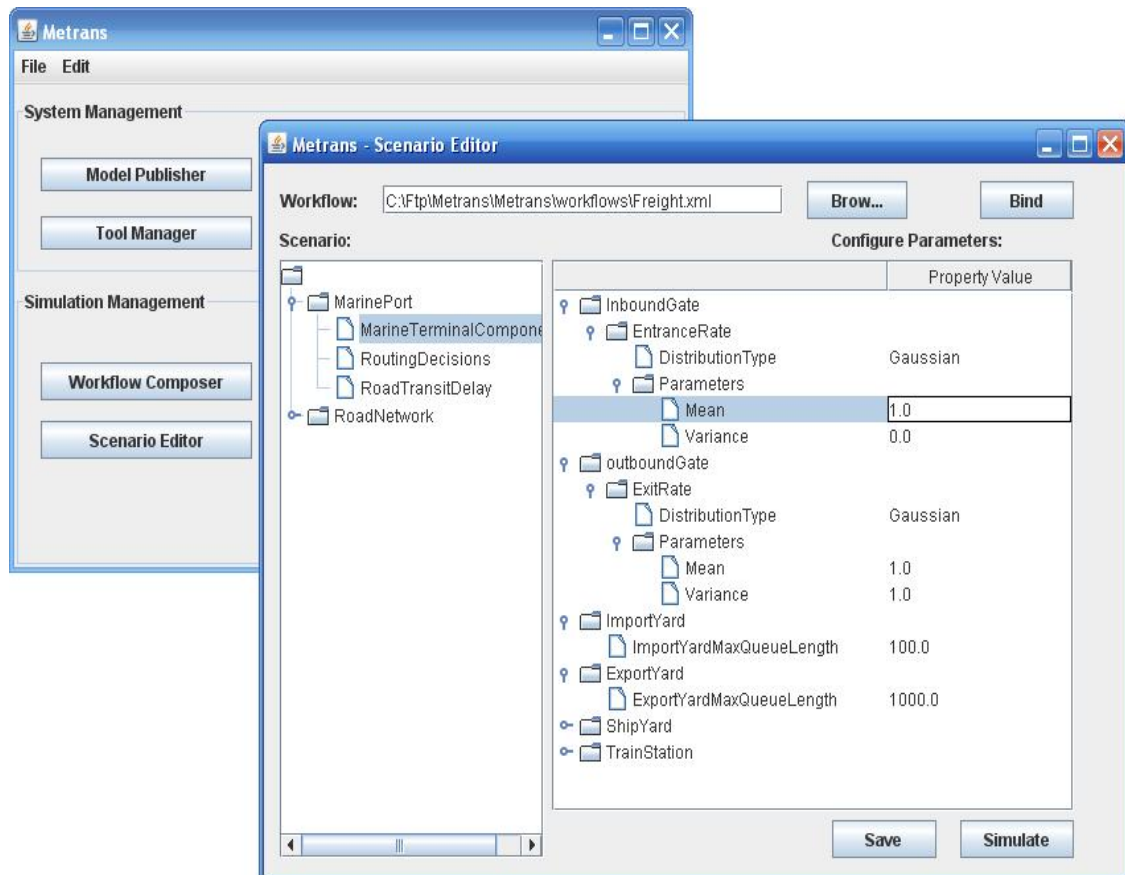


Figure 4-3: Scenario editor user interface

Domain experts create integrated simulation models through *model publisher* and *scenario editor* modules of the user interface. The *model publisher* is used to generate and store ontology-based simulation models. To enable integrated modeling, generic models designed using different simulation tools need to be translated to the ontology representations. For each integrated simulators, the publisher implements a model parser which is able to extract information from source simulation models and represent it using the ontology vocabulary. Base models are then stored in the metadata repository for later referencing. Model management including searching, viewing and retrieval are achieved using the SPARQL query language. The *scenario editor* as shown in Figure 4-3 allows domain experts to load ontology models from metadata repository and configure “what-if” simulation scenarios through a unified interface in the context of decision making. A scenario intrinsically represents an integrated simulation model, which is able to capture complex traffic systems by composing models defined using different simulators.

4.3 Workflow-based Application Integration

Simulation of transportation systems captured by integrated models need seamless integration of underlying software applications on a collaborative computing platform. In our vision, the technical requirements for application integration are as follows:

- It can activate and control execution engines without the help of Graphic User Interfaces of specific tools, and can advance the simulation according to a given simulation cycle.
- At the intermediate states, which are the states at the end of every simulation cycle, it can access data objects produced by integrated applications.
- It can model and automatically handle interactions such as data exchange pattern and execution logic between integrated applications.

Traditional approaches for traffic simulation integration were accomplished in a point-to-point fashion which is relatively static, inflexible, and is not scalable. Interactions between tools are typically hardcoded for specific simulation runs. Integrating new tools to such a system requires extensive system redesign and programming implementation. In our system, for each tool the framework implements a data adapter which is able to transform generic data produced by the tool into the general representation defined by the domain ontology model. Such data can be pushed to the metadata repository during simulation and is understandable across the system. Simulators and other computation tools read from the shared repository for data exchange. This hub-spoke communication architecture enables a modular and extensible integration framework.

Based on the communication platform, a workflow approach is used to provide flexible control and rapid execution of underlying tools. Workflow technology is an effective method to realize computing process control, process integration and process reconstruction. It has been used in many large-scale complex applications such as structural chemistry experiments [24], biological experiments [25], network simulation [26] etc. Workflow is the depiction of a group of activities organized by control paradigms, declared as the work of a person, work of a simple mechanism or work of a group of machines, software applications etc. It provides the glue for distributed applications and services. In our framework, simulation execution patterns are modeled as abstract workflows using a XML schema (e.g., execution sequence, simulation time step for each simulator and input/output relationships are modeled in the schema). Domain experts can customize the execution and data flow of underlying tools by configuring the workflow paradigm. Configured user workflows will be interpreted and translated to low-level workflows which can be executed by the simulation engine to actually run the simulations.

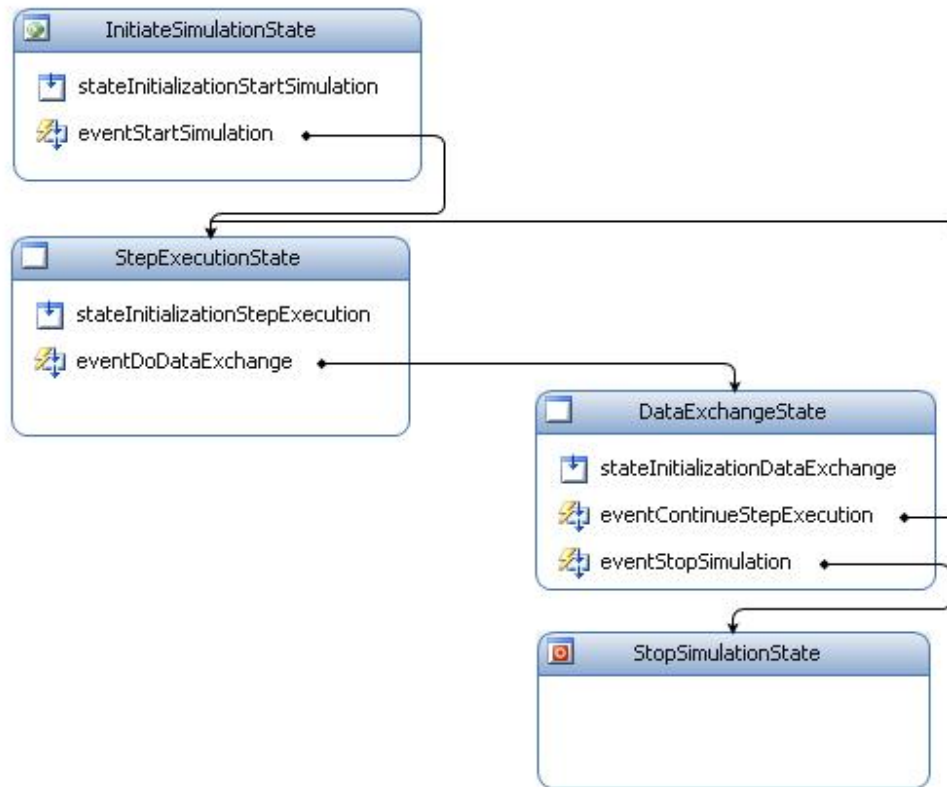


Figure 4-4: Simulation workflow paradigm

Various executable workflow languages/paradigms like BPEL [27], Xscufl [25], Windows Workflow Foundation (WF) [28] and Kepler [29] have been proposed and successfully used. Among them, many characteristics of WF make it an attractive choice to be used in our framework. These include the ability to embed workflow engine/runtime in a host application and to control and monitor the workflow status which makes it possible to write advanced workflow management environments which builds on the basic functionality framework. For example, in our case it is possible to embed the simulation engine, which is a WF workflow host application, within the environment used to define the high-level user workflows. The simulation engine is a simple state machine WF workflow application. It consists of four states including “initiate simulation”, “step execution”, “data exchange” and “stop simulation: as shown in Figure 4-4 and supports interactive execution of multiple simulation tools. Low-level details of execution control and data transform are hidden from end users and managed by the workflow runtime.

5. Empirical Evaluation

Management of integrated simulation is complex and tedious especially for domain experts such as control theorists and has become a barrier to high-quality research. In order to certify that the proposed framework is effective for integrated traffic simulation and control systems design, we studied a set of practical problems using the framework. The framework provides an intuitive environment to integrate computation tools and facilitates users to focus on investigating domain solutions instead of addressing software integration. It allows users to examine designs by defining and executing a set of “what-if” simulation scenarios (Figure 4-3).

We do not use quantitative measures to evaluate the performance of the integrated modeling and simulation framework. Instead, we validate the correctness of our framework by solving the same problems studied by existing integration approaches. We validate its usability by demonstrating intuitive steps taken to perform integrated simulations. We also validate the extensibility of our framework by showing that simulation applications are integrated on a modular basis. In the following, we use two representative examples to evaluate and highlight the utility of our framework.

In [2], researchers developed a simulation test bed that allows users to investigate the impact of various terminal operation strategies on the terminal capacity and cost as well as on the traffic network outside the terminals in an integrated manner. In their approach, two types of simulators including the VISSIM network simulator and TerSim terminal simulator are integrated. The execution flow and data communication between VISSIM and TerSim are hardcoded. On the other hand in [30], the researchers tightly coupled a highway ramp control algorithm with the VISSIM simulator to study the effect of their ramp controller. We replicate the above two integrated simulations using our framework and show the common VISSIM network simulator can be shared.

5.1 Marine Terminal – Traffic Network Simulation

We employed the integration framework to study freight transportation near a marine terminal from the LA Long Beach port as shown in Figure 5-1. The problem is to maintain high terminal throughput while at the same time manage congestion and maintain traffic efficiency in the traffic network outside the terminal. This by itself is very challenging due to multiple factors that affect each mode of freight transportation as well as the complex interactions between different modes of transportation. Neither a terminal model nor a traffic network model can be used alone to evaluate new strategies

and make decisions. For instance, a model of a marine port may be used to predict an increase in terminal throughput as a result of a change in cargo handling strategies or introduction of a new technology. However, unless this evaluation includes a simulator that models the entire transportation chain, such an evaluation may easily ignore the fact that improved terminal throughput could cause an increase in congestion on the traffic network used by trucks to serve the terminal. To study this problem, integrated modeling and simulation of the entire transportation chain is needed.

In our experiment, we assume the marine port is modeled using TermSim and the traffic network connected to the terminal is modeled using VISSIM. Compared to [2], the major steps taken to study the target freight transportation systems are as follows:

Publishing base models: Domain scientists design base terminal models and traffic models using TermSim and VISSIM respectively. These models can be re-configured or modified to represent new concepts or operational/control strategies of the modeled sub-systems. The generic terminal and network models are published to the integration environment, translated to ontology representations and stored in the common repository so that they can be used to compose integrated freight transportation systems.

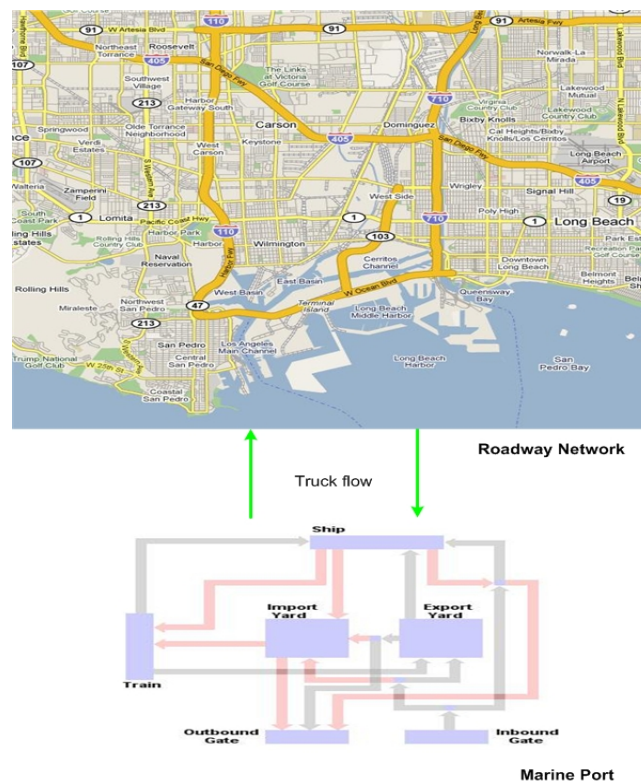


Figure 5-1: Terminal – traffic network freight transportation Scenario

Integrating simulation tools: TerSim and VISSIM are both microscopic, time step simulators. Figure 5-2 shows the execution logic of the integrated simulation which was also adopted in [2]. In each cycle, the two simulators execute for a given time step and then exchange data. The simulation stops when the total simulation time is reached. As we observed, this type of execution logic is popular in integrated simulations and can be realized by using the four-state state machine WF workflow we introduced in Section 4.3. We implemented the data adapters for TerSim and VISSIM respectively to enable data transforms between them and the ontology model. Users can merely select the WF workflow paradigm, configure the simulation time step, input/output relationships of the terminal and traffic network models (e.g., inbound or outbound gate of the terminal is connected to a specific road of the traffic network) to realize the same integrated simulation in [2]. Low-level details of execution control, data transform etc. are handled by the workflow engine automatically.

Design space exploration: Domain experts can configure and simulate a set of intuitive “what-if” scenarios to evaluate their designs in either terminal or traffic network system. The framework enables users to bind the pre-configured execution workflow to a scenario, load network models and terminal models to the *scenario editor* and customize simulations in an integrated manner (Figure 4-3). Various terminal operational strategies and local traffic conditions (e.g., increasing the processing rate of inbound/outbound gates, changing the routing decisions of trucks outside the terminal etc.) can be modeled and the impact can be evaluated intuitively.

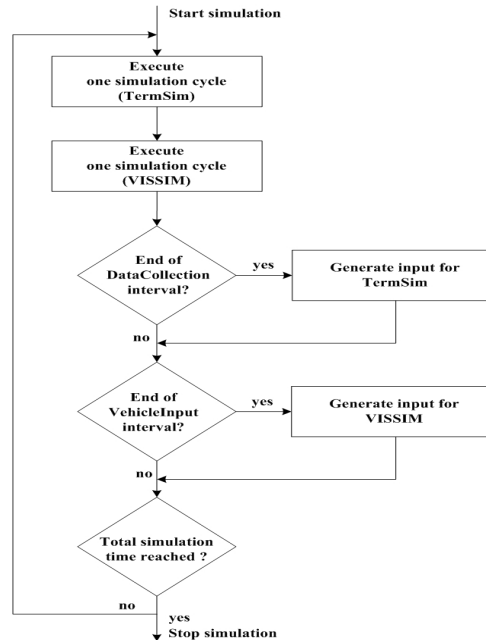


Figure 5-2: Control logic of integrated TerSim-VISSIM simulation

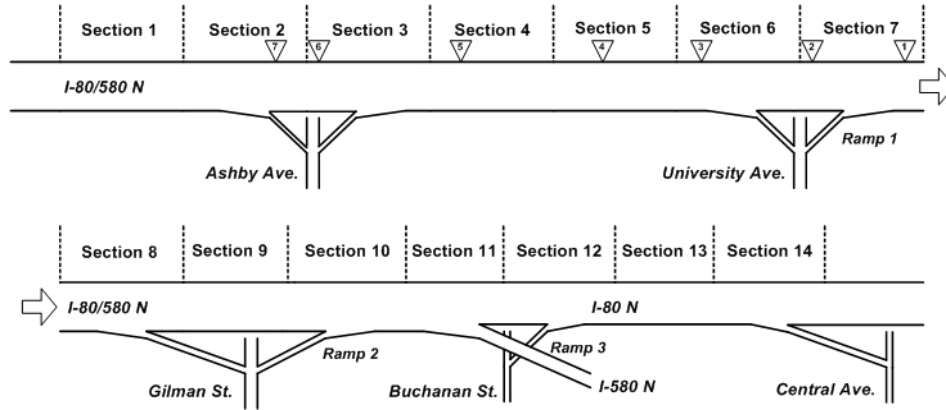


Figure 5-3: Layout of the freeway model

5.2 Highway Ramp Control Simulation

Our framework can also be used to conduct simulations to evaluate generic traffic control algorithms. As an example we employed the integration framework to evaluate the highway ramp controller proposed in [30]. The control system that integrates ramp metering and speed limits is designed to relieve congestion, suppress shock waves and improve safety on a freeway stretch. The algorithm is implemented using Matlab and the highway is modeled using VISSIM.

In this case, VISSIM is integrated with the Matlab program. Matlab is one of the most widely used tools for control system design and simulation. Providing the framework support the integration of Matlab with simulation tools is especially useful for transportation research. In this scenario, at each simulation cycle the roadway controller takes the vehicle flow rates as inputs and calculates the metering rates and speed limits for the highway. The execution logic can also be modeled by the four-state state machine workflow of our simulation framework. At the step execution state, the simulation engine advances the VISSIM highway simulation and calls the Matlab algorithm to compute the ramp control and highway speed limit for the next cycle. At the data exchange state, the simulation engine accesses the VISSIM kernel functions to collect the vehicle flow rates, transform and load the data to the Matlab workspace, and sets ramp control signal and speed limit parameters in VISSIM simulation environment.

Table 5-1: Ramp control simulation results

scenario	stdK	stdV	TTS1	TTS2	TTS3	TTSA
1	7.4	26.2	295.6	99.0	62.8	457.4
2	7.1	26.1	298.8	99.0	55.3	453.1

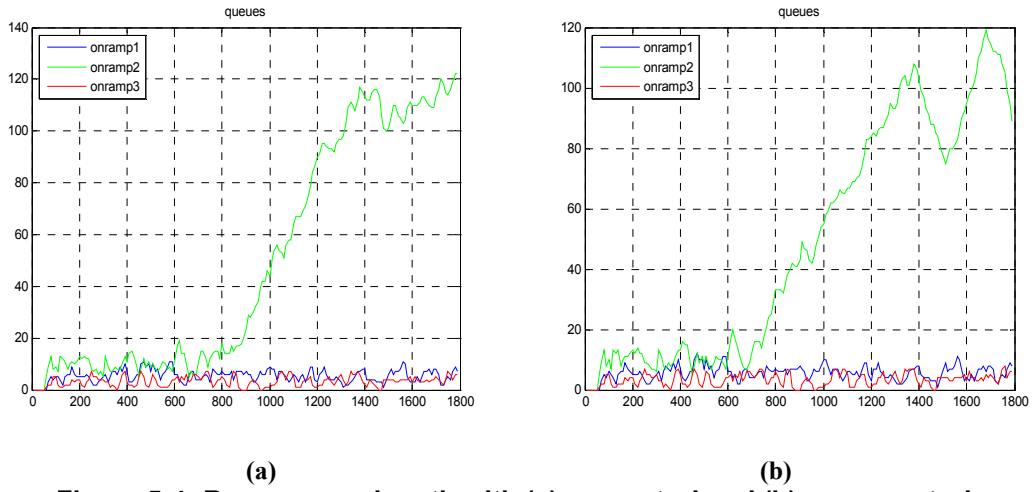


Figure 5-4: Ramp queue length with (a) no control and (b) ramp control

We follow the three intuitive steps described in 5.1 to evaluate the performances of the highway ramp controller, except we can reuse the VISSIM simulator which was already integrated in the framework. We studied the same problem as studied in [30]. Assuming there is a disturbance on the freeway (see Figure 5-3) at section 11 which is near to ramp 2, two scenarios are simulated with 1) no control; 2) applying the ramp metering controller. Figure 5-4 shows the queue length at ramp 2 for both cases and Table 5-1 shows the simulation results. The total travel time (TTSA) for case 2 is less than that in case 1 and the average queue length at ramp 2 for case 2 is smaller than that in case 1. From the results we can conclude the controller is able to reduce the freeway congestion. The impact of the roadway control is not significant in this case, our simulation framework enables users to optimize the algorithm by interactively tuning the controller and traffic flow parameters through the integrated scenario editor.

6. Conclusion

In this project we developed an integrated traffic modeling and simulation framework that allows us to investigate the impact of various technologies and control strategies on freight transportation systems. The integration system is of general use and could be employed to evaluate a wide range of concepts, technologies and configurations associated with the scenarios. The framework is modular and extensible, researchers developing simulation models, control algorithms using different tools etc., is able to insert new modules into the environment with little effort and without modifying existing modules.

The key components of the framework include the domain ontology, the metadata repository and the workflow-based simulation engine. The domain model provides a common modeling language and forms the conceptual basis for data and application integration. The metadata repository provides model addressing and acts as a shared medium for data exchange which reduces the complexity of passing heterogeneous data among underlying modules. The workflow-based simulation engine provides fine grained execution control of simulation applications. Compared to traditional approaches for integrated traffic simulation, the proposed framework realizes tool integration in great ease and allows intuitive design space exploration.

7. Implementation

In this project we developed an integrated traffic modeling and simulation framework that allows us to investigate the impact of various technologies and control strategies on freight transportation systems. Our implementation of the integration framework is based on Windows operating system with .NET framework 3.0 and java JDK 6.0.

As examples, freight transportation in a terminal from the LA Long Beach port with associated traffic network outside the terminal and ramp control of highway sections were studied. Related simulation and computing tools including TerSim, VISSIM and Matlab were integrated to demonstrate the use of the test bed for integrated transportation modeling and simulation.

Simulation tools that our framework has integrated, for example, the TerSim and VISSIM simulators, are assumed to be traffic simulators which support necessary runtime expansion features. These tools should provide interfaces to enable the integration framework import/export data and control simulation steps at runtime. Since the data exchange between simulation tools are based on a none point-to-point architecture, users only need to implement data adapters which map tool-specific traffic models with the common ontology representation to enable integration. Any changes in the structure of the data collection and I/O functions of one tool will not require other modules in the system to be recoded.

8. References

- [1] Arnaud Doniec, Stéphane Espié, René Mandiau and Sylvain Piechowiak, "Multi-agent Coordination and Anticipation Model to Design a Road Traffic Simulation Tool," *4th European Workshop on Multi-Agent Systems*, 2006.
- [2] Ioannou P., Chassiakos A., Valencia G. and Hwan C., "Simulation Test-bed and Evaluation of Truck Movement Concepts on Terminal Efficiency and Traffic Flow," *Metrans Project 05-11 Final Report*, 2007.
- [3] http://www.atac.com/Products_Airports-a.html
- [4] Buisson C., Lebaque J.P. and Lessort J.B., "A Discretized Macroscopic Model of Vehicular Traffic Flow in Complex Networks Based on the Godunov Scheme," *Proceedings of CESE'96*, 1996.
- [5] Elloumi N., Haj-Salem H. et al, METACOR: "A Macroscopic Modeling Tool for Urban Corridor," *Proceedings of Triennial Symposium on Transport Analysis*, 1994.
- [6] Jayakrishnan, R., Mahmassani H.S. and Hu T.Y., "An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Networks," *Transportation Research C*, 1994.
- [7] Ben-Akiva, M., et al., "Real-time Simulation of Traffic Demand-supply Interactions within DynaMIT," *Transportation and network analysis: current trends*, 2002.
- [8] http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl
- [9] <http://mctrans.ce.ufl.edu/featured/tsis>
- [10] Calvanese, D., De Giacomo G., Lenzerini M., et al, "Information Integration: Conceptual Modeling and Reasoning Support," *3rd IFCIS Intl. Conf. on Cooperative Information Systems*, 1998.
- [11] Burghout W., Wahlstedt J., "Hybrid Traffic Simulation with Adaptive Signal Control," *Transportation Research Record*, 2007.
- [12] Sahraoui A., Jayakrishnan R., Microscopic-macroscopic Models Systems Integration: a Simulation Case Study for ATMIS," *Simulation*, 2005.
- [13] Piorkowski M., Raya M. and Lugo A.L., et al, "TraNS: Joint Traffic and Network Simulator for VANETs," *Proceedings of the 13th ACM MobiCom*, 2007.
- [14] Ramakrishna S., Bakshi A., Prasanna V.K., "A Semantic Framework for Integrated Asset Management in Smart Oilfields," *Proceedings of the 7th IEEE International Symposium on Cluster Computing and the Grid*, 2007.
- [15] Wache H., Vogeles T., Visser U., et al, "Ontology-based Integration of Information: A Survey of Existing Approaches," *Proceedings of IJCAI-01 Workshop: Ontologies and Information Sharing*, 2001.

- [16] Qunzhi Zhou, V. K. Prasanna, Hwan Chang and Yun Wang, "A Semantic Framework for Integrated Modeling and Simulation of Transportation Systems," *12th IFAC Symposium on Control in Transportation Systems*, Los Angeles, 2009.
- [17] Qunzhi Zhou, A. Bakshi, V. K. Prasanna et al, "Towards an Integrated Modeling and Simulation Framework for Freight Transportation in metropolitan Areas," *IEEE International Conference on Information Reuse and Integration*, Las Vegas, 2008.
- [18] <http://www.w3.org/2001/sw/>
- [19] Tran Quoc Dung, Kameyama, W., "A Proposal of Ontology-based Health Care Information Extraction System: VnHIES," *IEEE International Conference on Research, Innovation and Vision for the Future*, 2007.
- [20] Van Niekerk J.C., Griffiths K., "Advancing Health Care Management with the Semantic Web," *3rd International Conference on Broadband Communications, Information Technology & Biomedical Applications*, 2008.
- [21] Taswell C., "DOORS to the Semantic Web and Grid With a PORTAL for Biomedical Computing," *IEEE Trans. on Information Technology in Biomedicine*, Vol. 12, Issue 2, March 2008.
- [22] Cong Z., V. K. Prasanna, et al, "Data Component Based Management of Reservoir Simulation Models," *IEEE International Conference on Information Reuse and Integration*, 2008.
- [23] <http://www.w3.org/TR/owl-features/>
- [24] Baldrige K.K., Greenberg J.P., Sudholt W., Mock S., et al, "The Computational Chemistry Prototyping Environment," *Proceedings of the IEEE*, Volume 93, Issue 3, 2005.
- [25] Oinn T., Greenwood M., Addis M., Alpdemir N. et al, "Taverna: Lessons in Creating a Workflow Environment for the Life Sciences," *Concurrency and Computation: Practice and Experience*, Vol. 18, No. 10, 2006.
- [26] Eric Eide, Leigh Stoller, et al, "Integrated Scientific Workflow Management for the Emulab Network Testbed," *USENIX Annual Technical Conference*, 2006.
- [27] F. Curbera, Y. Golland, J. Klein, F. Leymann, D. Rollerand, and S. Weerawarana, "Business Process Execution Language for Web Services, version 1.1. specification," May 2003.
- [28] P. Andrew, Presenting Windows Workflow Foundation Beta Edition. SAMS Publishing.
- [29] B. Ludascher, K. Lin, S. Bowers, E. Jaeger-Frank, B. Brodaric, and C. Baru, "Managing Scientific Data: From Data Integration to Scientific Workflows," *GSA Today, Special Issue on Geoinformatics*, 2005.
- [30] Hwan Chang, Yun Wang and Ioannou P., "An Integrated Roadway Controller and its Evaluation by Microscopic Simulator VISSIM," *Proceedings of ECC'07*, 2007.