AN OBJECT-ORIENTED APPROACH FOR THE ARCHITECTURE DESIGN OF THE MANAGEMENT OF NARROW PASSAGEWAYS

Evangelos I. Kaisar, Ph.D. Candidate, kaisar@wam.umd.edu
Mark Austin, Associate Professor austin@eng.umd.edu
Ali Haghani, Professor and Chairman haghani@eng.umd.edu
University of Maryland
Department of Civil and Environmental Engineering
1173 Glen Martin Hall
College Park, MD 20742
USA

ABSTRACT

Narrow passageways are a significant source of traffic congestion and delay in transportation networks. With traffic volumes expected to increase significantly in the foreseeable future, the effective management of these passageways is needed to mitigate the undesirable impact of these bottlenecks on transportation system safety, performance and cost. In an effort to address the significant challenges associated with the analysis, design, and implementation of appropriate management operations for narrow passageways, an object-based model for the management of narrow passageways in the transportation network is developed. The methodology proposed in this paper is a first step toward end-to-end synthesis and validation of narrow passageway of transportation systems. We formulate a methodology that combines ideas from object-based and systems-engineering development,and propose a step-by-step procedure for transforming operations concepts into a system-level design, and validating behavior via formal modeling procedures.

INTRODUCTION

In transportation networks, narrow passageways occur where the width of a transportation link is insufficient to permit operation of two-way traffic at normal speeds of operation. As a result, congestion, accidents, and delays are common problems. The effective management of these systems is needed to mitigate the undesirable impact of bottlenecks on system safety, performance and cost (Lagakos et al. 2001). Examples of narrow passageways occuring in transportation networks are: waterway, work zone, tunnel, one-lane bridge, and railroad applications. While the management of traffic on a one-lane bridge might be handled with a sign indicating who has the right of way, the implementation of appropriate management operations for large scale transportation systems can be very complex and expensive, in part, because modern communication systems must be integrated with automated scheduling, surveillance and tracking systems.

In an effort to address the significant challenges associated with the analysis, design, and implementation of appropriate management operations for narrow passageways, the first goal of this paper is to develop an object-based model for the management of narrow passageway problems in transportation systems. The object model is developed in two steps. First, we identify high-level management functionality, objects, and associated data/information sources that are common to all narrow passageway applications. In the second step, functionality of the object model is customized (or extended) to the specific needs of the narrow passageway application domain (e.g., waterways and work zones). By creating a hierarchical object model for narrow passageway management operations, we hope that in the near-term, engineers will be provided with improved methods for analyzing the system behavior of complex management operations, designing and upgrading new systems through improved procedures for requirements generation, and reusing established system architectures and systems integration across application domains (e.g., decision making procedures for management operations guided by information sources obtained from Geographic Information Systems (GIS)).

Scope of Narrow Passageway Problems.

Figure 1 shows the scope of narrow passageway management operations that will be addressed in this study. The upper-most level represents management operations that are common to all narrow passageway problems. At the second level, management operations for specific narrow passageway problems is obtained through extension and specialization of the high-level generic management operations.

The object hierarchy shown in Figure 1 is derived from a wide range of real world

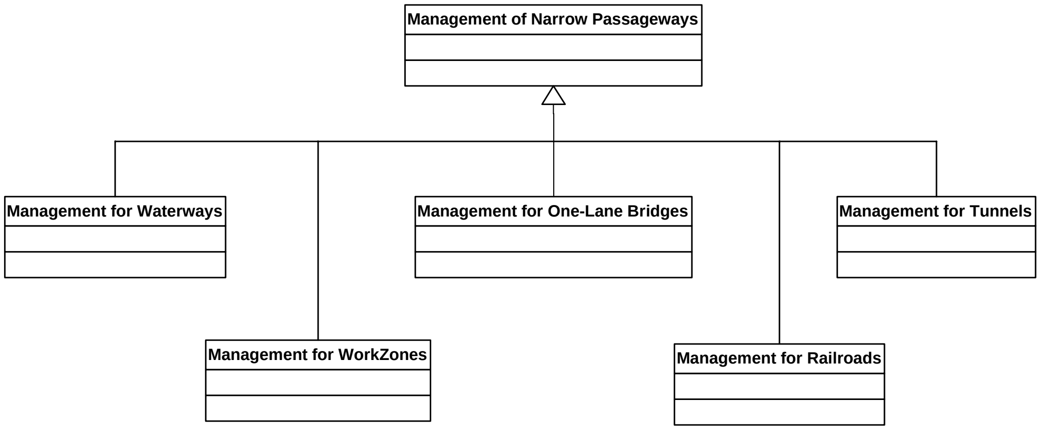

Figure1. Hierarchy of Management Problems.

transportation systems. For railroad, work zone and narrow waterway applications, sophisticated techniques of system analysis and control are justified by the life-critical safety risks and adverse economics of poor system throughput. As a case in point, university researchers have worked with the Federal Highway Administration to study optimal geometry and appropriate management system control policies for traffic flow in highway and urban street work zones (Schonfeld 1999). Even more sophisticated management operations are justified when high volume traffic streams need to transit narrow passageways embedded within large-scale transportation networks. For example, research (Dai 1998) has been conducted to understand and design traffic control policies for inland waterways containing locks (e.g., the Mississippi River, Danube River). Preventing accidents and environmental disasters, reducing congestion and lengthy traffic delays, enforcing laws, and collecting tolls are all essential tasks of a traffic management system.

Countries such as Panama and Turkey have already made large investments in the development of enhanced traffic management systems for narrow waterways. A notable application is the Bosporus Striaght, a sinuous 19-mile long straight with 12 abrupt turns and treacherous currents, that is a passageway for nearly 50,000 tankers and cargo vessels a year. It has become an artery for the world's oil because of the oil exports from the former Soviet republics. Traffic jams and shipwrecks have become quite common. In December of 1999, a Russian tanker split in half and polluted the water and coastline with 900 tons of fuel oil. In April of 1999, a 9,000-ton cargo ship ran out of control and crashed into the shoreline of Bebek, an Istanbul neighborhood on the Bosporus Straight. Alarmed by the growing safety, environmental and economic threats of the Bosporus's overcrowding, Turkey hired Lockheed Martin to build a $20 million system that will improve the control of ship traffic through the use of radar and satellite technologies (Moore 2000). Another notable application is the Gaillard Cut, which is an eight-mile narrow segment of the Panama Canal that can only support unidirectional traffic at any one time. The delays in transit service caused by this part of the canal lead to significant increases in fuel cost, service costs, and depreciation costs for vessels and their cargo. Delays may become even more significant as traffic demands continue to grow (Panama 1998). Moreover, due to the strategic nature of the Panama Canal as a transportation and trade link between the Pacific and Atlantic Oceans, and the strong need for expeditious transit service, the Panama Canal Agency is creating an enhanced traffic management system and widening the Gaillard Cut as part of a one-billion dollar canal improvement program (Panama 1998).

The traffic control management system that we are proposing includes a traffic management information system (TMIS) and effective communications between the control center and traffic streams within the narrow channel. The TMIS has at its disposal a variety of technologies for collection and transmission of data (e.g., cameras, sensors, GPS, radio communications). Control centers employ the TMIS to collect and process information sources, such as details on the position of traffic, weather, and safety information.

OBJECT-BASED/SYSTEM-BASED PROBLEM FORMULATION

With the need for a generic approach to modeling, design and management of narrow passageway problems in place, we formulate a methodology that combines ideas from object-based and systems-engineering development, and propose a step-by –step procedure for transforming an operation concept through a system-level design. We show that concepts of narrow passageway application areas can be organized into conceptual class hierarchies suitable for reuse.

We employ the Unified Modeling Language (UML), a collection of eight diagram types capable of

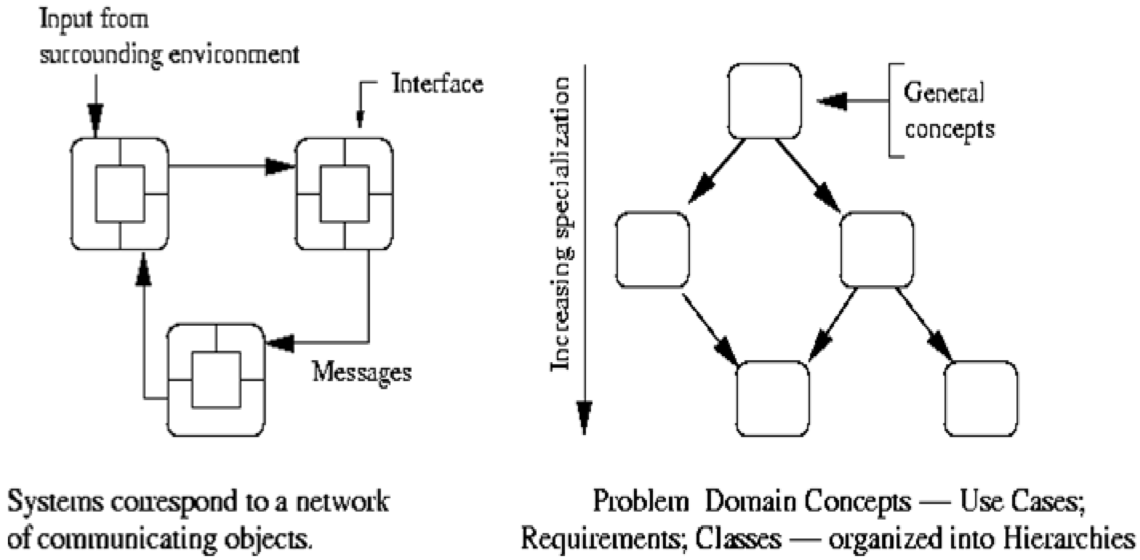

Figure 2. Key Concepts in Object-Oriented Development (Austin 2004).

modeling transportation system behavior and structure. Key aspects of front-end development include a use-case model for a visual representation of high-level management functionality and a domain model for building an understanding of the domain in which the management system is being developed.

Object-Based Models

Ideas in object-oriented development have been around since the late 1960s. The object-oriented paradigm (i.e., way of doing things) is motivated by a need to: (1). Simplify the way we view the real world, and (2). Provide engineers with mechanisms for handling complex problem that are subject to change. Figure 2. illustrates two key elements of object-oriented development. On the left hand side, object-based systems correspond to networks of communicating objects and systems. They achieve their purpose with modules having having well defined functionality, well defined interfaces for connectivity to other modules and the surrounding environment, and message passing. Once object-based models have been formulated, problem domain concepts are organized into class hierarchies for reuse. Designers need to identify objects and their attributes and functions, establish relationships among the objects, establish the interfaces or each object, implement and test the individual objects, assemble and test the system, and organize classes for reuse (via persistent storage in a database).

System Development Process

Complex narrow passageway systems are much more than "just a collection of objects." To minimize the possibility of unforeseen failure we need models of system-level development that will help designers clearly articulate what the system must provide and what must be prevented. Engineers also need to understand the extent to which a system provides functionality beyond what is actually required. As shown on the left-hand side of Figure 3, the development process for object-based modeling begins with the construction of a use case model, containing use case diagrams representing high-level system functionality. In this initial phase of life cycle development, high-level requirements are gathered through goals and scenario analysis (Lagakos 2000). An event table can help designers identify expected results to each case and events that are initiated by external agents or actors that are shown in the use case. Second, a domain model consisting of a conceptual static structure diagram and collaboration or sequence diagram is constructed. Together these diagrams and models portray the real world domain in which the system is being developed. In this analysis phase, conceptual class diagrams are useful for representing the system structure. Collaboration and sequence diagrams show the flows of communication among objects needed to support required system behavior. System design alternatives are created by mapping models of system behavior onto the system structure. Finally, statechart and deployment diagrams are useful representations for detailed system-level design, subsystem design, and implementation – these latter stages of development are beyond the scope of this paper, however.

Complicated scenarios can be organized into hierarchies of activity and sequence diagrams. For each task, a sequence diagram can show the components and sequence of messages that would implemented the system functionality. We create the system-level by mapping fragments of behavior onto the system structure (part of this mapping process is containeed in the sequence diagrams in the previous step) and imposing constraints on performance and

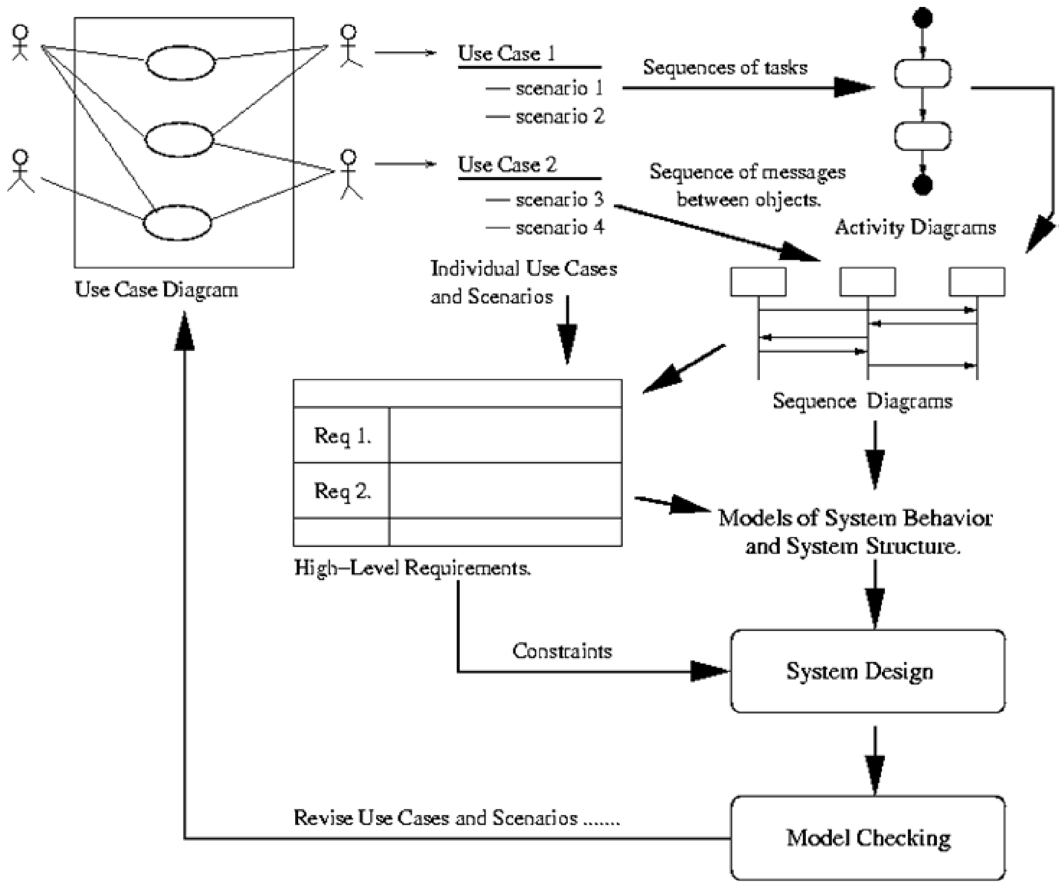

Figure 3. Step-by-Step Procedure for Synthesis and Validation of System Level Designs for Management of Narrow Passageways (Austin 2004).

operation. Downstream, these fragments of behavior will be composed together to form models of system-level behavior. In addition, we have to do model checking. Model checking procedures make sure that the system design (1) does what it`s supposed to do; (2) prevents certain behaviors from occuring; and (3) does not support un-intended behaviors. If any one of these aspects is violated, then we have a “gap” between the intented system and the actual system design. We can close gaps in the system design by refining the scenarios. This, leads to more detailed diagrams and a modified system-level architecture.

Use-Case Model

A use case is simply a set of system scenarios tied together by a common user goal (i.e., aspect of system functionality). A use case specification contains:

- A list of actors (actors are anything that interfaces with the system externally);
- A boundary separating the system from its external environment;
- A description of information flows between the actors and individual use cases;
- A description of normal flow of events for the use case, and
- A description of alternative and/or exceptional flows.

Use case diagrams are a convenient way of showing the way in which a real-world actor will interact with the system, the use cases with which they are involved, and the boundary of the application.

A collection of use cases is known as a use case model (Ambler 1998). The development of a use case model provides order to the elicitation and representation of high-level system functionality, which in turn, leads to the generation of requirements, identification of system objects and their interaction.
Figure 4 is a use case diagram for a "general purpose" traffic management system. The names of the actors, which are drawn as stick figures, are control center, traffic controller, driver, and country/authorities. At the heart of the traffic management system is the control center. The control center monitors and tracks traffic in passageways to maintain safety, ensure security and law enforcement, protect the environment, while also scheduling and optimizing traffic operations. It monitors weather conditions and traffic congestion, and dispatches quick-response units to respond to emergencies. The major points of

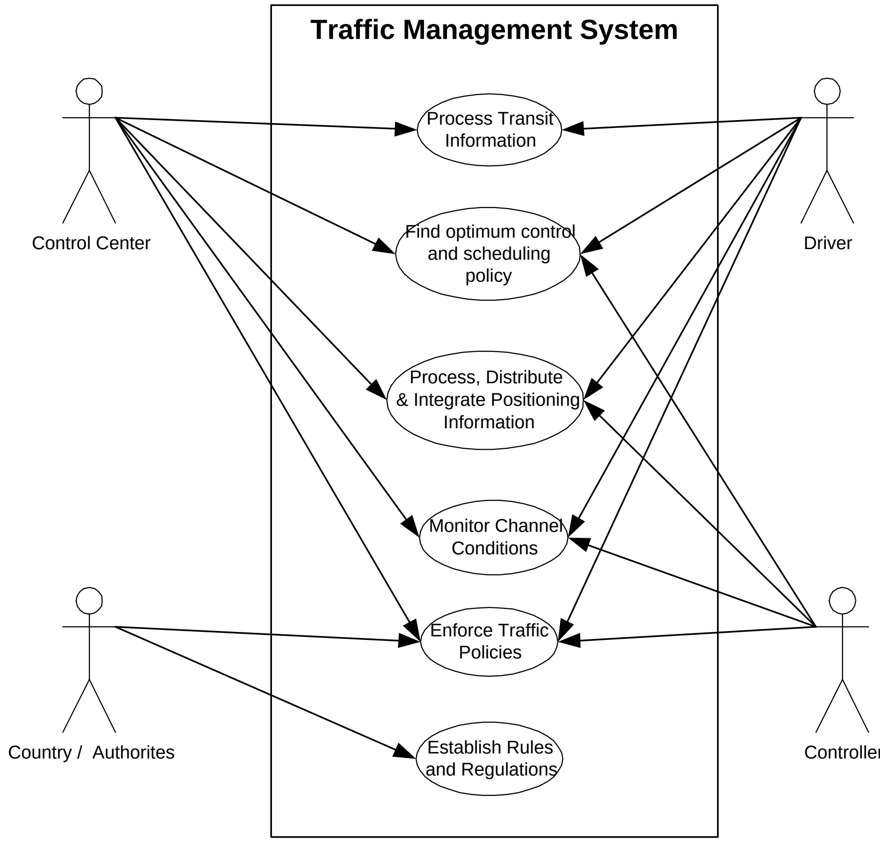

Figure 4. Use Case Diagram of a High-level Traffic Management System.

contact for a control center are:

1. The traffic controllers, who implement the traffic policies imposed by the control center. Controllers are physically located at the narrow passageway and can either be humans or automated devices. For example, lock operators and tugboats can be controllers at inland waterways and canals while traffic lights, automated switches and other electronic devices can be controllers at tunnels, one-way bridges and railroads.
2. The authorities, who enforce the rules of passageway operation and respond to an emergency (for this problem domain, a country is a political establishment that establishes rules and regulations, and in some cases tolls).
3. The drivers, who transit the narrow passageways. Drivers receive route and departure time information from the control center aimed at avoiding congested passageways. So-called "top-of-the-line" Traffic Management Information Systems would also allow drivers to send positioning information and traffic updates to the control center.

The use case diagram does not indicate the objects and flow of data/information in the transportation system. The structure of the system and the information flowing into and out of the system is not identified until the next phase of modeling represented by the domain model. The use case diagram does not indicate the objects and flow of data information in the transportation system. The objects and data flow are later displayed in domain model diagrams. Nor does the use case diagram display the expected results for each use case. An event table is used to indicate the system's response to each event that is indicated by external agents or actors in each use case scenario.

Event Table Model

An event table list possible events in rows and important information regarding each evet in column. Each column in event table is descibed below:

- Event: The event that activates the system.
- Trigger: An occurrence of information that is inputed into the system.
- Source: An external agent or actor that initiates the trigger and provides the input data into the system.
- Activity: The action that the system performs in response to the event trigger.
- Response: The output that is produced by the system.
- Destination: An external agent or actor that receives the output response from the system

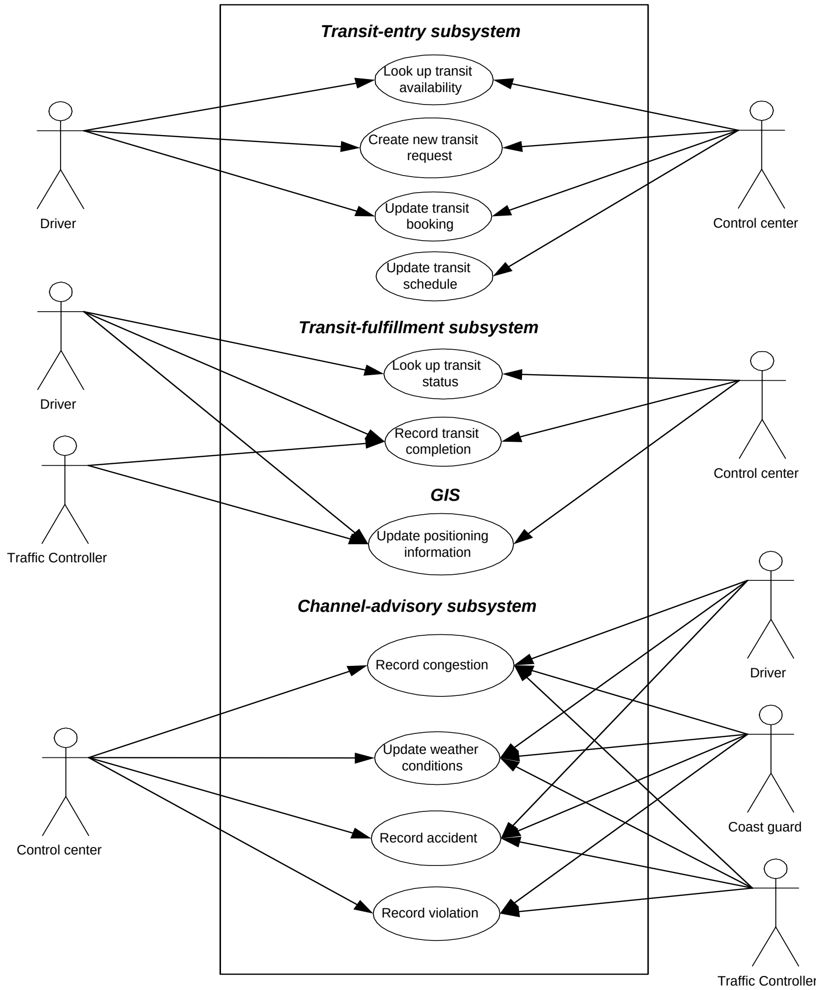

Figure 5. Use Case Diagram for Management of Waterways.

The above columns in the event table help describe how the system reacts to. The event table is a convenient way to record key information about the requirements for the information system. It is used to create object oriented models for the management of narrow passageways like waterways and workzones. Examples of event tables are later shown in the waterways application domain.

Domain Model

With the use case model in hand, the purpose of the domain model is to build an understanding of the problem domain in which the system is being developed. Figure 6 shows the objects that make up the high-level system structure -- rectangles represent the various classes and the roles they take within the application, and the lines between the classes represent the relationships or associations between them together with their multiplicity. UML notation allows for the representation of a variety of multiplicity relationships. For example, every route has at least one passageway and every passageway can belong to one or more routes. In this case there is a one-to-many relationship in both directions.

The class diagram can be customized (or extended) to class diagrams for any narrow passageway application domain. Figure 7 shows, for example, extension of the high-level class diagrams to traffic management systems in waterways. The new extended class inherits all the atributes and methods from the high-level class.

APPLICATION TO TRAFFIC MANAGEMENT IN NARROW WATERWAYS

The objective of the object model formulation for the management of narrow passageway problems is to customize it to the specific needs of the narrow passageway domain such as waterways. In order to meet these specific needs, requirements of the narrow waterway application domain are gathered by expanding the use cases in Figure 4 to more custom made use cases in Figure 5. The actors who interact with the specialized system and the high-level functionality are the same in both use case diagrams. However, as expected, management of narrow waterways requires more specialized functionality as in the case of transit entry and fulfillment subsystems. While these subsystems may require more complex

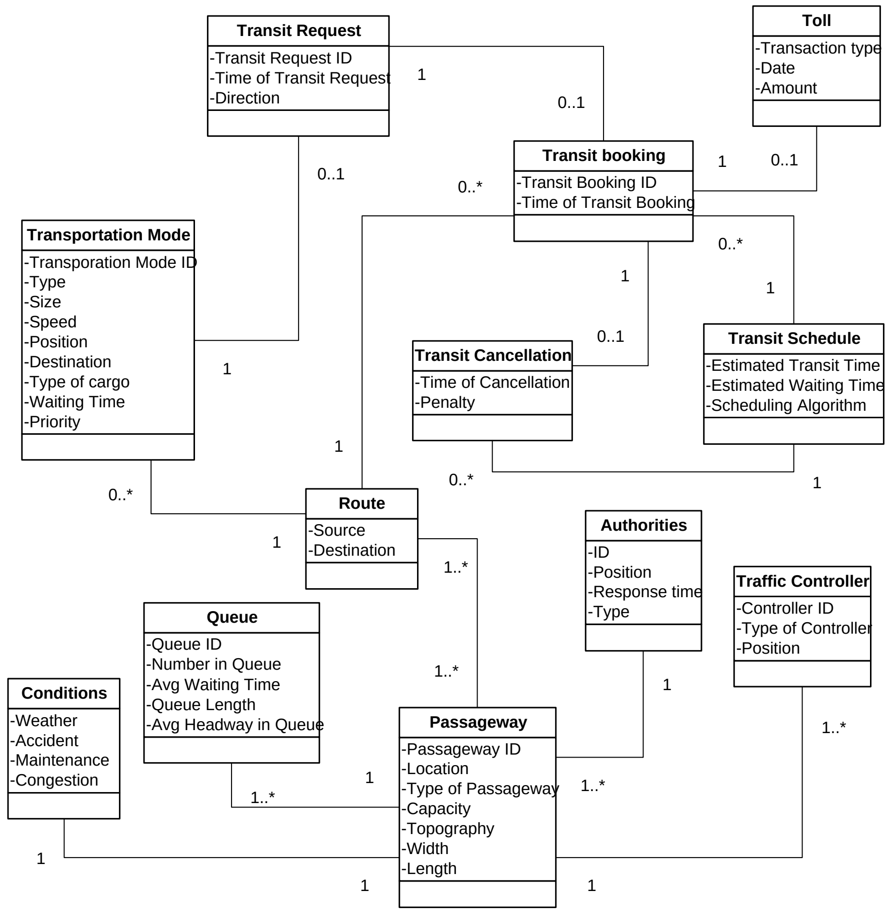

Figure 6. Conceptual Class Diagram for High-Level Traffic Management System

operations and information systems than similar systems in other transportation networks, the use case model in the object-based formulation can be applied to the management of narrow waterways. However the actors in this application initiate a different set of events that are listed in the brief event table in Table 1. In this Table the events: captain completes transit, captain sends positioning information, queue has too many vessels, bad weather in the channel, accident in the channel, and vessel violated rules have been discarded. As expected these events lead to more specialized functionality for the management of narrow waterways as in the case of transit entry and fulfillment subsystems.

The structure of the waterway problem domain is also derived from the high-level conceptual class diagram for narrow passageways. Figure 6 shows the conceptual class diagram for a management system in the waterway application. Classes such as the vessel and waterway are extensions of the transportation mode and passageway, respectively. The vessel class inherits all the attributes of the transportation mode while including additional information such as the attributes, vessel type and nationality. The specific element (vessel) is fully consistent with the general element (transportation mode in this case).

Another relationship that is specific to the waterway application is that every vessel makes one transit request while in the general case in Figure 4 it is not mandatory for a transportation mode to make a transit request as in the case of cars passing through a work zone. The fact that all vessels make transit requests to pass through the channel, makes the transit booking and scheduling ever so important for a traffic management information system (TMIS) in the waterway application.

TMIS Subsystems

The transit-entry subsystem has use cases for looking up transit availability, creating a new transit request, updating a transit booking and updating the transit schedule. This subsystem processes transit inquiries, requests and cancellations from drivers who call the control center. A transit request may be made as the vessel approaches the narrow channel or beforehand and there may or may not be a toll associated with the transit. Once a transit request is received and a booking is made, the transit schedule is automatically updated by an algorithm that will

Event	Trigger	Source	Activity	Response	**Destination**
Captain or Agent checks for transit availability.	Transit inquiry	Captain Agent Wireless or Terrestrial network	Look up transit availability	Transit availability details	Captain/Agent
Captain makes a request for passage	New transit request	Captain	Create new transit request and update transit schedule	Transit Booking/Details Transit schedule Transaction/Toll	Captain Control Center Bank
Captain changes or cancels transit booking	Transit change request	Captain	Update transit booking and schedule	Change confirmation Transit change details Transaction Penalty	Captain Control Center
Captain/Agent wants to check transit status	Transit status inquiry	Captain/Agent	Look up transit status	Transit status details	Captain/Agent

Table 1. Abbreviated Event Table for Management of Waterways.

help reduce delays and congestion and the transit information is sent to the driver and traffic controller at the waterway.

Transit scheduling and traffic control policy can provide for more efficient transits through narrow waterways, especially when the narrow waterways are embedded within a large-scale waterway network. For example, a vessel may be redirected to an alternate route in order to avoid a congested waterway. Providing an alternate route based on a distance vector or link state routing algorithm (Huitema 1995) can help reduce transit times at congested waterways. Also control policies such as dispatching disciplines at waterway locks can also help provide better transit system performance. For example, Ting and Schonfeld used simulation to analyze different dispatching disciplines such as “first come first serve” (FCFS) and “shortest processing time first” (SPF) at a series of waterway locks in the Mississippi River. They concluded that SPF was more preferable to the FCFS dispatching discipline for reducing delays at each lock. Simulation is a powerful tool for determining the optimal transit route and control policy (Ting 1998) and the object model approach is a powerful method for understanding the requirements, the structure and the behavior of a system.

The Geographic Information System (GIS) produces maps with the position of all the objects in the waterway channel (see Figure 5). This integrated map is sent to traffic controllers and drivers who are in transit. The ability to locate all the objects in the channel provides for additional safety and more feature with an accurate representation of its banks, braids, and navigable channels on the river. Finally, a river could be modeled as a sinuous line forming a trough in a surface model. From the river’s path through the surface, the information system can calculate its profile and rate of descent, the watershed it drains it drains and its flooding potential for a prescribed rainfall (Evans 1993). Using GIS to model the narrow waterway and locate the position of all ships on a map, can help the control center, traffic controllers and drivers make intelligent decisions and manage the most difficult areas of navigation efficient operations. In addition, the GIS can be used to model the actual channel in different ways so that it easier to manage. For example, a GIS can model rivers as a set of lines that form a network. A linear network model can then be applied to analyze ship traffic. A river could also be modeled as an aerial.it drains and its flooding potential for a prescribed rainfall (Evans 1993). Using GIS to model the narrow waterway and locate the position of all ships on a map, can help the control center, traffic controllers and drivers make intelligent decisions and manage the most difficult areas of navigationefficient operations.

Finally, the Channel Advisory Subsystem sends updates on channel conditions to operators, controllers, drivers and the coast guard. Congestion, inclement weather, accidents and violation of rules and regulations all require special attention. An information system can keep track of such conditions and alert all the actors of these special circumstances. Such an advisory system can help provide safe and efficient transits for all ships.

VALIDATION AND VERIFICATION

The terms system validation and verification refer to two basic concerns, "are we building the right product" and "are we building the product right?"

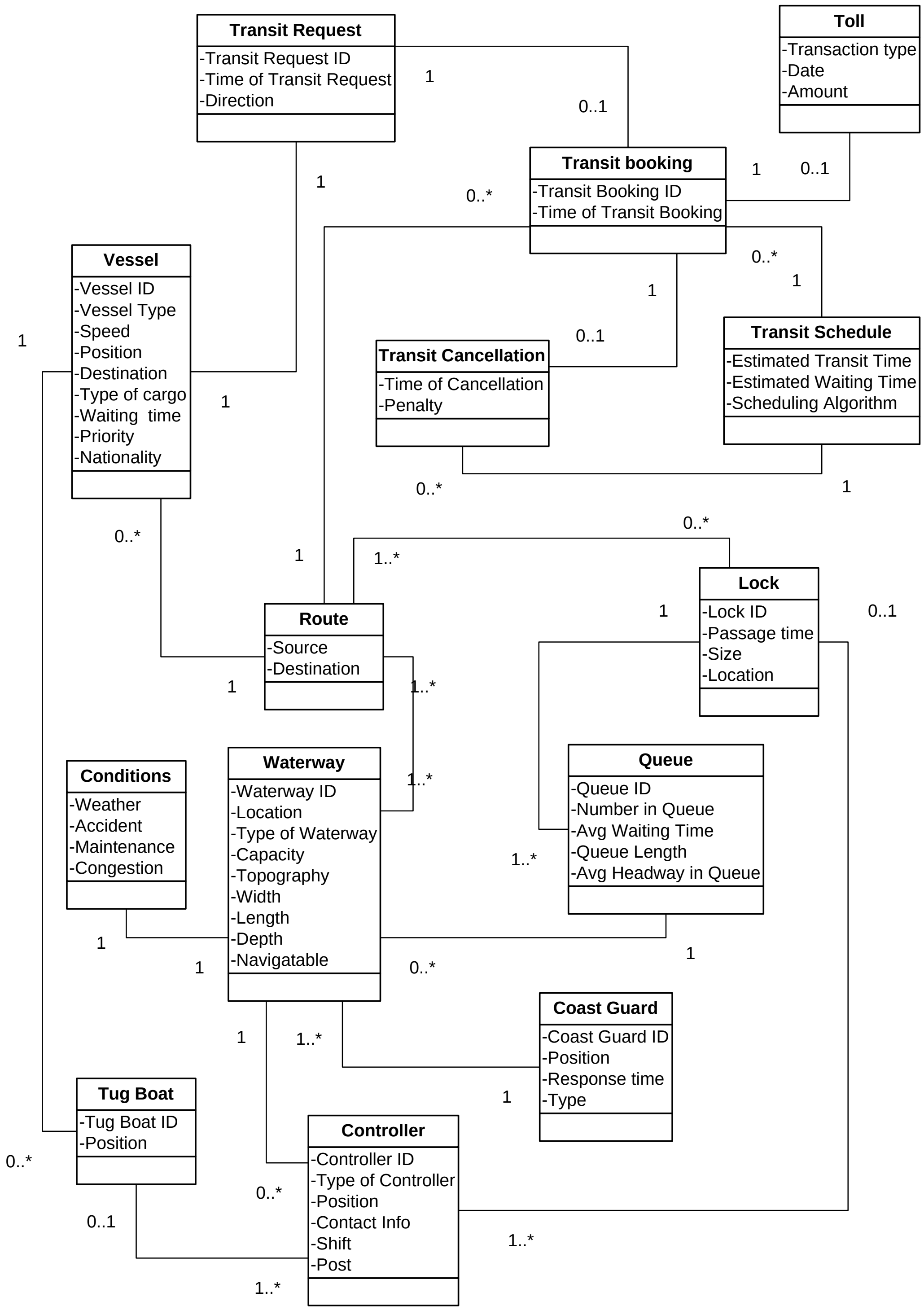

Figure 7. Conceptual Class Diagram for a Traffic Management System in Waterways.

Satisfactory answers to both questions are a prerequisite to customer acceptance. Verification is simply a process of determining when the system's components will meet their requirements. In a historical sense, validation procedures have beenviewed as something that are executed near the end of project completion. Now that systems are become more heterogeneous and complex, there is now a general move toward application of validation procedures throughout the system life cycle process. Engineers need to check consistency between:

(1) the stakeholders' needs and the and operations concept,
(2) the operations concept and the initial (originating) requirements, and
(3) initial and derived requirements.

Consistency must also be maintained between the various layers of abstraction and specification (e.g., system; subsystem; module; component) that are produced (Larsen 2002).

The "model checking" task shown at the bottom of Figure 2 corresponds to formal models of the system-level design. We assume that system architectures will be represented as networks of communicating finite state machines. We need to examine traces of message passing and communication to make sure the system will do what it's supposed to. A good system: (1) exhibits safety and liveliness and (2) avoids deadlocks. From a design perspective, we are particularly interested traces that are exhibited by the architecture model, but have not been specified as a desirable aspect of system functionalisy in the use case/scenario analysis.

By detecting and validating implied scenarios, it is possible to drive the elaborarion of scenario-based specifications and behavior models to a converged state. The iterative procedure that results is shown along the bottom of Figure 2. Formal procedures and models for detecting implied scenarios can be found in the work of Uchitel and co-workers (Uchitel 2003; LTSA 2003).

CONCLUSION

The object-oriented model views a system as a group of interacting objects that work together to accomplish system objectives and satisfy system requirements. Key benefits in the object approach to problem solving include:

1. Reuse of high-level system architectures across narrow passageway applications.
2. Ease of extension to specific application domains, like waterways, work zones, railroads, tunnels and one-way bridges.
3. Representation and solution of problems of a relatively high level of abstraction.

Together, these benefits improve problem solving productivity. For front-end development, use-case and domain models provide a visual representation of high-level system functionality and system design. Engineers can use this methodology for behavior analysis of complex operations, to design and upgrade new systems, and to reuse and integrate geographic information systems, transit scheduling, and channel advisory information systems. We anticipate that a TMIS integrated with state-of-the-art technologies like GPS, cameras, sensors and radio communications will lead to improvements in transit time, throughput, and reductions in cost associated with delays and accidents. These advances will be of great use to countries, organizations, and companies that are currently developing and investing in very complex and expensive traffic management systems.

Our medium-term research plans are to automate the development process by designing and building an interactive problem solving environment for the front-end development of surface transportation systems. Research tasks include finding ways to store UML diagrams in object-relational databases (see, for example, Smallworld, 2000) and connect front-end developments/graphics with high-level back-end systems analysis and rule-checking procedures. In addition providing preliminary feedback on system performance, we hope that the latter will define boundaries between the feasible and infeasible domains, and simplify problem formulations by representing (and possibly eliminating) technology options and management procedures early in the development lifecycle.

REFERENCES

Ambler, S., 1998, "*How the UML Models Fit Together*". Software Development.

Austin, M., 2004. "*An Introduction to Information - Centric System Engineering*". Tutorial F06, Toulouse, France, (20-24 June), INCOSE.

Dai, M. D., and P. Schonfeld. 1998. "*Metamodels for Estimating Waterway Delays Through Series of Qeues*". Transportation Research, Part b,32(1),1-19.

Evans,T.A.; D. Djokic, and D. Maidment, 1993. "*Development and Appplication of Expert Geographic Information System*" Journal of Computing in Civil Engineering, Volume 7, No. 3, pp.339-353.

Huitema, C. 1995. "*Routing in the Internet*".

Moore, Moley. 2000, "The Bosporus: A Clogged Artery" .*Washington Post*, A29, November 16.

Labeled Transition System Analyzer (LTSA), 2003. http://www.doc.ic.ac.uk/~jnm/book/ltsa/LTSA.html.

Lagakos, V.; E. Kaisar, and M. Austin, 2000. "*Object Modelling for the Management of Narrow Passageways in Transportation Systems*". Proceedings of the International Council of System Engineering Conference, Melbourne, Australia, (July), INCOSE.

Larsen R.F. and D.M.Buede, 2002. "*Theoretical Framework for the Continuous Early (CEaVa) Method*". Systems Engineering, Vol. 5, No. 3, pp. 223-241.

Panama Canal Commission.. 1998, *Annual Report.*

Schonfeld, P. and S. Chien, 1999. "*Optimal Work Zone Lengths For Two-lane Highways*" Journal of Transportation Engineering, Vol 125, No. 1.

Smallworld Database (2001). See: http://www.gesmallworld.com

Ting, C. and P. Schonfeld, 1998. "*Integrated Control for Series of Waterway Locks*" Journal of Waterway, Port, Coastal, and Ocean Engineering, Vol 124, No. 4.

Uchitel S. 2003. "Incremental Elaboraation of Scenario-Based Specifications and Behavior Models using Implied Scenarios". Ph.D. Thesis, Imperial College, London.