Use of CORBA and Object Oriented Concepts in the Gary-Chicago-Milwaukee (GCM) Gateway Traveler Information System

David Zavattero
ITS Program Manager
Illinois Department of Transportation
120 West Center Court
Schaumburg, Illinois 60195
Tele: 847-705-4800
Fax: 847-705-4803
Email: zavatteroda@nt.dot.state.il.us

Wei Wu
Pr. Transportation Systems Engineer
Parsons Corporation
525 West Monroe Street, 10th Floor
Chicago, Illinois 60661
Tele: 312-930-5115
Fax: 312-930-0018
Email: Wei.Wu@parsons.com

Paper prepared for the 2003 Annual Meeting of the Transportation Research Board
Use of CORBA and Object Oriented Concepts in the Gary-Chicago-Milwaukee (GCM) Gateway Traveler Information System

David A. Zavattero, Illinois DOT and Wei Wu, Parsons Corporation

Abstract

The Gary-Chicago-Milwaukee (GCM) Corridor is one of the four priority corridors established under the ISTEA legislation of 1991. The Gateway TIS is the core system that facilitates the integration and interoperability of the many Intelligent Transportation Systems in the Corridor. The Gateway TIS collects transportation related information from geographically dispersed agencies of varied natures, validates and fuses the information collected, and disseminates it to interested public and private entities and the general public via internet and intranet.

The foundation for the Gateway architecture is the Common Object Request Broker Architecture (CORBA). CORBA is one of the two NTCIP Center-To-Center (C2C) communication protocols. CORBA supports both Center-to-Center (C2C) communications and internal, distributed, object oriented computing. CORBA is rapidly gaining popularity and momentum in the current software engineering trend of object orientation and distributed computing. It is likely that CORBA will become a popular solution to both C2C communication and distributed computing for next generation ITS. The experience of the Gateway TIS in implementing a CORBA based solution for C2C communications and internal computing can provide useful insights to other ITS center systems currently being planned, designed, and implemented.

This paper presents a high level view of the system architecture design of the GCM Gateway TIS with a focus on ITS C2C interoperability. The object oriented Gateway external data model and the Gateway Publisher/Subscriber based data collection and distribution are also introduced. The decision to utilize an object oriented Database Management Structure (DBMS) versus a relational DBMS is also discussed.
Use of CORBA and Object Oriented Concepts in the Gary-Chicago-Milwaukee (GCM) Gateway Traveler Information System

David A. Zavattero, Illinois DOT and Wei Wu, Parsons Corporation

BACKGROUND
The Gary-Chicago-Milwaukee (GCM) Corridor is one of the four priority corridors established under the ISTEA legislation. The Gateway Traveler Information System (TIS) is the core system that facilitates the integration and interoperation of the many Intelligent Transportation Systems (ITS) within this corridor. The Gateway TIS collects transportation related information from geographically dispersed systems of varied natures, validates and fuses the information collected, and disseminates it to interested public and private entities and the general public via Internet and Intranet.

The Gateway TIS is designed to handle a wide range of traffic related information, including both incidents and planned events that affect traffic operations; status and data from field devices such as vehicle detectors, cameras, variable message signs, highway advisory radios, and weather sensor stations; as well as derived traffic parameters such as congestion level, travel time and speed. While no transit information is served in its initial phase, the future phases of the Gateway will provide greatly enhanced transit information. The Gateway interfaces with a variety of ITS related systems for data collection and distribution purposes, including traffic management centers, departments of transportation, transit operators, emergency dispatch centers, police and fire departments, weather systems, and value added traveler information service providers.

The foundation of the Gateway architecture is the Common Object Request Broker Architecture (CORBA) for both its Center-to-Center (C2C) communications and internal inter-process communications. CORBA is an industry standard developed by the Object Management Group (OMG) for object-oriented distributed computing. CORBA has been adopted as one of the two National Transportation Communication for ITS Protocol (NTCIP) standards for C2C communications. However, none of the existing transportation domain standards needed for C2C applications, such as Location Reference Message Sets (LRMS), Traffic Management Data Dictionaries (TMDD) for Advanced Traveler Information Systems (ATIS) and Advanced Traffic Management systems (ATMS), or Message Sets for External Traffic Management Center Communications (MS/ETMC2) are intended for object-oriented software engineering procedures. The Gateway system design has pioneered the effort to develop a full set of CORBA-based corridor-wide standards to support C2C ATIS applications. The CORBA-based standards developed for the Gateway project are based on the relevant non-CORBA standards.

The Gateway system is currently undergoing intensive system integration and testing and is scheduled to be online and operational in 2002. As CORBA is rapidly gaining popularity and momentum in the current software engineering trend of object orientation and distributed computing, it is expected that CORBA will be a prominent solution to both C2C communication and distributed computing for many new generation ITS deployments. The experience of the Gateway TIS provides useful insights to other ITS center systems that are being planned, designed, and implemented. The CORBA based transportation domain standards developed by the Gateway for the GCM Corridor can serve as a field proven starting point for the needed
This paper presents the high level architecture of the Gateway system, followed by a discussion of major system wide design issues such including: the use of CORBA, the Gateway Data Object Model (GDOM) defined in CORBA Interface Definition Language, and the Publisher/Subscriber based Gateway Message Oriented Middleware (GMOM). The process that led to the selection of an object-oriented database management system for the Gateway implementation is briefly discussed. This paper also provides an introduction to the Gateway website and serves as a call for the development of the CORBA oriented specifications of national ITS domain standards essential to achieving true CORBA based C2C interoperability.

THE GCM CORRIDOR

The GCM corridor encompasses the major urban areas in three states, includes 16 counties, extends 130 miles, covers more than 2,500 square miles, and is home to more than 10 million people and 5 million jobs. The GCM corridor also includes major air and freight hubs serving the regional, U.S., and world economies. Since its designation as a Priority Corridor in 1993 a wide variety of ITS projects have been implemented by various agencies. The GCM Multi-Modal Traveler Information System (MMTIS) Project [1] developed a multi-tiered, corridor-wide traveler information system architecture. At the center of this architecture is the corridor’s traveler information hub: the Gateway Traveler Information System (TIS). The ultimate Gateway TIS consists of, and connects, a central, multi-state Gateway Hub and three state/regional Gateway Hubs for information exchange. In turn, each state/regional hub interfaces with TIS related agencies within their jurisdictions for data collection and re-distribution purpose. In addition, the central Gateway Hub and the state/regional Gateway Hubs also provide the infrastructure for coordinated operation of ITS infrastructure such as joint device control should this capability be desired and needed in the future.

Figure 1 illustrates the planned GCM Corridor TIS architecture which include a central Gateway hub and three state/regional hubs: the Illinois Hub for northeastern Illinois, the Borman ATMS Hub for northwestern Indiana, and the CDSI (Computer Data Systems Infrastructure) Hub for southeastern Wisconsin, referred to as the Illinois Gateway, Indiana Gateway, and Wisconsin Gateway hubs respectively. Implementation of the corridor architecture is incremental. At this point in time, the Illinois and Wisconsin Gateway Hubs are under system definition and development, and the Illinois Gateway Hub functions are co-located with the central Gateway Hub. As a result, the initial phase central Gateway Hub interfaces with individual ITS components directly rather than through the state/regional hubs.

The design and implementation of the initial phase Gateway TIS started in July 1998 and will become operational in 2002. The Gateway TIS replaces the current Corridor Traveler Information Center (CTIC), which has been providing traveler information services in the corridor since mid 1990s. Along with the software development, a Wide Area Network (WAN) for the corridor has been defined and is being developed. The ultimate corridor WAN will have an ATM (Asynchronous Transfer Mode) fiber optic network connecting the central Gateway Hub, the state/regional Gateway Hubs, and the major intelligent transportation systems operating in the three-state corridor (such as regional traffic management centers, transit control centers, and emergency service centers). ATM technology was selected because of its inherent capabilities for integrating the broad array of video, data, and voice communications needed to serve the corridor operators and customers.

The Gateway architecture also calls for a Gateway Local Area Network (LAN). The
ultimate Gateway LAN will be an Ethernet 100BaseT copper network which operates in parallel to the optical OC-3/OC-12 ATM network. A group of Sun Microsystems servers are deployed to provide high performance, heavy-duty, back-end data processing and web page services. In the front-end, several Intel-processor based workstations are used as operator and supervisor workstations. The Gateway software architecture makes extensive use of CORBA [2] for its C2C communications and its internal, distributed, object oriented computing.

An Objected-Oriented Database Management System (OODBMS) was selected for the Gateway because of its superior performance over Relational Database Management Systems (RDMS). While C++ is used where performance is critical such as the database interface, Java is used as the main programming language largely for its platform independence, web integration readiness, availability of extensive class libraries, and ease of implementation.

GATEWAY SYSTEM ARCHITECTURE AND SUBSYSTEMS
At the highest level, the Gateway consists of ten subsystems. Figure 2 provides a logical overview of the system architecture. A brief description of each individual subsystem is provided in the following sections. Additional information is available through the GCM Public Information Center (see the www.travelinfo.org website).

Data Source Interface Subsystem (DSI)
The DSI subsystem is responsible for interfacing with external data source systems to receive data; filter private, sensitive and irrelevant data; convert data to the Gateway standard format if needed; and send the data via the CORBA ORB (Object Request Broker) to the Gateway’s central site. If the data source system is a CORBA enabled system, these interfacing functions can be performed by the source systems internally. Otherwise, an interfacing machine needs to be used. A channel based Publisher/Subscriber (P/S) implementation is run at either the data source side or the Gateway central site to facilitate the delivery of data to the Gateway. The P/S system is a complex construct that is used extensively by the Gateway TIS. A detailed discussion of the P/S system is provided as a separate section.

Data Acquisition Subsystem (DAQ)
The DAQ is responsible for receiving data from data source interface subsystem and passing the data to the Data Validation and Fusion Subsystem for further processing. Acting as a subscribing client of a P/S system, the DAQ subsystem subscribes to various data sources with specifications of the types of data the Gateway intends to receive. The relevant data is then forwarded to it by the P/S system whenever the data becomes available.

Data Validation and Fusion Subsystem (DVF)
The DVF subsystem is responsible for validating data of all kinds, integrating data from various sources over space and time, performing traffic status analysis, saving data to the database, and routing data to the Data Distribution Subsystem (DDS). Data validation includes both simple sanity checks and complicated location and identity checks. The location check ensures that locations reported with different location referencing methods can be uniquely and unambiguously resolved to the underlying Gateway map database. The Gateway supports location-referencing methods using geographical coordinates, cross street, mile-markers, addresses, segment identifiers, and a limited set of landmarks. Ultimately, all LRMS (Location Referencing Message Specifications) methods will be supported by the Gateway. The identity check is to minimize the possibility that an event reported by multiple sources would appear in
the Gateway as separate events. Routing data directly to the Data Distribution Subsystem (DDS) while it is being saved to the database reduces the intensity of database assessing operations required by the DDS while maintaining a higher data concurrency for the data distribution. The DVF subsystem ensures that data distributed by the Gateway is as accurate, comprehensive, and timely as possible.

**Data Storage Subsystem (DSS)**

The DSS is responsible for managing and facilitating the interactions between all the other Gateway subsystems, the database, and the database management system. In addition to the common database functions, the Gateway database also provides a large array of Geographic Information System (GIS) capabilities needed by TIS operations, such as location resolution, location referencing method translation, and spatial data fusion.

The DSS has three layers. At the core is an object-oriented Database Management System (DBMS) that provides the very high performance needed by the Gateway TIS. A separate section will address the DBMS selection in detail. The middle layer is a large, developed C++ library built upon the DBMS API (Application Programming Interface) to provide an assortment of specialized database operations. To support distributed database access and database access by multiple programming languages such as Java and C++, a CORBA wrapper is used as the third layer to interface between all database clients and the underlying database management system.

**Data Distribution Subsystem (DDS)**

The DDS is responsible for distributing validated and fused data to the other Gateway subsystems and external systems. It is the focal point for the dissemination of Gateway data commonly needed by Gateway external data users and by some internal Gateway subsystems such as the Graphical User Interface and Web Server subsystems. The core of the DSS is a facility to maintain the current state of the corridor wide traveler information and a data distribution channel that publishes real-time data to its subscribers. Though needs for database access by subsystems other than the DDS remain, the DDS minimizes the overall database access demand by maintaining the current state of the database locally.

**Web Server Subsystem (WSS)**

The WSS is responsible for creating all web pages to be shown on the Gateway web site over both Internet and Intranet. The Gateway web server offers pages openly accessible to the general public as well as pages with various access restrictions intended for protected access by authorized public and private entities. Both interactive graphic maps and textual data are available in the HTML (HyperText Markup Language) format. JavaServer Pages (JSP) and JavaScript technologies are used to provide rich dynamics. In the future, XML (Extensible Markup Language) web pages will also be offered through the Gateway.

**Monitoring, Logging, Notification Subsystem (MLN)**

The MLN is responsible for monitoring and logging the status of the Gateway system, taking rectifying actions when system elements fail, and notifying appropriate persons of abnormal situations. The status of all objects, processes, resources, network connections, and machines of interest to the Gateway are monitored, whether they are at the central Gateway Hub, at the state/regional Gateway Hubs, or at the data sources.
Security Subsystem (SS)
Gateway security covers four facets of operations: Encryption, Authentication, Authorization and Auditing. Encryption ensures that data between two parties remains private and that no one can eavesdrop on their communications. Authentication ensures that only the proper users are accessing the system. Authorization ensures that each user is allowed to access only those parts of the system they are authorized to access. Auditing allows a user’s actions to be traced. The SS is responsible for providing for Authorization and Authentication checking. The Encryption and Auditing security functions are implemented by the operating system, purchased software, or other Gateway systems.

Administration Subsystem (ADM)
The ADM subsystem is responsible for providing tools and utilities for performing administrative tasks, such as user management, system backup and recovery, system log management, life cycle management of system processes, and network and resource management.

Graphical User Interface Subsystem (GUI)
The GUI subsystem is responsible for providing a set of user-friendly graphical interfaces for Gateway operators and supervisors, to perform the intended functions such as:

- Enter, list, update, or cancel traffic related events
- Monitor traffic status and traffic related events in both tabular and map based graphical formats
- Perform basic system administration
- Monitor system health

SYSTEM WIDE DESIGN – USE OF CORBA
The Gateway uses CORBA extensively for both its communications with external TIS related systems and for its inter-process communications. CORBA is one of the two C2C protocol specifications being developed under the NTCIP standards process [3]. Compared to the socket-based DATEX specification, CORBA is a new and very powerful technology. Not many ITS center applications are currently using CORBA in their system design and planning. Fewer are implementing CORBA technology for both C2C and inter-process communications. But, CORBA is rapidly gaining popularity and momentum in object-oriented software engineering practices for a wide range of distributed computing information systems. As high-speed computing hardware and high band-width communications become more affordable and available, it is likely that the inherent advantages of CORBA will make it a preferred solution to both C2C communication and distributed computing for many new generation ITS systems.

Choice: CORBA vs. DATEX
The use of CORBA was the most important system-wide decision made in the Gateway design. The alternative DATEX [3] is a communication stack approach, and was initially proposed as a protocol specification because it is a natural extension of the NTCIP Class C protocol. DATEX tries to build interoperability into the NTCIP stack by standardizing as many communications protocol layers as possible. The problems associated with this approach were clearly identified by the C2C Working Group which noted “extending the Class C Protocol to handle the multitudinous heterogeneous systems that could typically make up a regional corridor, would be impractical. It would require all layers of the communications stack to be specified and a large
number of services would have to be developed on all of the participating systems to ensure interoperability.” CORBA takes a drastically different approach to address these problems through the provision of an object oriented application layer that takes full responsibility for interoperability by building into the upper stack layer all the conversions necessary to meet the requirements for semantic, service, syntactic, and session interoperability. Through CORBA the Gateway system provides a framework for the integration and interoperability of the many ITS systems and varied technical platforms that exist in the GCM Corridor. CORBA offers a superior solution for this environment.

The C2C Working Group also identified concerns in using CORBA in time critical missions such as near-real-time device control (on the order of once-a-second) and in integrating “legacy systems”. Latency of delay does not constitute a problem for the functions the Gateway is intended to provide. However, the Gateway does have to integrate with a large number of external systems that may not be using CORBA. The Gateway design solves this problem by deploying an interface computer to act as a bridge between the Gateway and the legacy systems. The interface computer performs conversions of communications protocols and data formats.

CORBA also has several additional advantages over DATEX. CORBA provides a number of essential services such as naming, event, security, and trader services that have not been specified under DATEX to date. For example, the CORBA Naming Service allows users to associate meaningful names with objects, which can subsequently be used by other objects to locate the named objects in a lookup facility. The CORBA Event Service allows the de-coupling of the event producers (data originators) and event consumers (data users). This capability serves as the basis for the Gateway Message Oriented Middleware (GMOM) described below.

**Center-to-Center Interface**
The Gateway C2C interface is the focal point where data exchanges between the Gateway and other GCM TIS related systems occur. The design of this interface is critical to achieving corridor wide interoperability and integration among ITS related systems. The Gateway C2C interface must be an open, flexible, scalable, standards compliant architecture to interface with the growing number of ITS related systems of varied technical capabilities and platforms. With these considerations strongly in mind, the Gateway C2C interface has been designed to consist of two principal components: the Gateway Data Object Model (GDOM) and the Gateway Message Oriented Middleware (GMOM). The GDOM defines the format of data. The GMOM defines the how data is transferred.

**Gateway Data Object Model (GDOM)**
The Gateway Data Object Model defines the data format for data exchange with other systems. Since no CORBA-based national ITS data standards were available at the time of the Gateway design, it was necessary to develop a set of CORBA based C2C ATIS data standards to apply the CORBA technology to the Gateway system. A strong objective was to achieve consistency with existing and emerging national non-CORBA standards. An intensive study of relevant national data standardization efforts being undertaken for non-CORBA based applications and a careful evaluation of the Southern California Priority Corridor Showcase Project High Level Design: Early Start Kernel System Requirements [4] was undertaken as the first step in the development of the GDOM. National standardization efforts including the Traffic Management Data Dictionary (TMDD) [5], Message Sets for External Traffic Management Center Communication (MS/ETMC2) [5], and the Location Reference Message Specification (LRMS) [6] were also considered in the development of the GDOM.
The GDOM is defined in IDL (Interface Definition Language). As a result, the basic data elements, which are equivalent to data elements in the TMDD, are strongly typed, and independent of hardware platform, operating system, and application programming language. The use of IDL allows the creation of an object oriented data model. Unlike the MS/ETMC2 and LRMS standards that specify flat message structures, the GDOM exploits the advantages offered by object oriented technology such as encapsulation, inheritance, and polymorphism.

A detailed description of the GDOM is provided in the Gateway External IDL Interface Document available through the GCM Public Information Center [1].

Gateway Message Oriented Middleware (GMOM)
The GMOM is a Publisher/Subscriber (P/S) system built on the CORBA Event Services. A P/S system allows a source of data (publisher) to simultaneously send an event to multiple users (subscribers). In the simplest P/S implementation, all users interested in receiving data register with the data source, and the source sends data to each registered user. This requires that the source and user know about each other. A more sophisticated implementation of P/S uses a mediator (channel) where a separate server acts as a go-between. Both the source and user register with this mediator. The source publishes data to the mediator and the mediator then forwards the data to all subscribing users. The Gateway implements a mediator based P/S system. The major services provided by this P/S system include:

- Discovery – which allows the subscriber to inquire and discover a list of available data publication services, i.e. to find out what publications are available.
- Type Subscription – which allows the subscriber to subscribe to a type of publication that will include the publications of its subtypes, e.g., to subscribe to the Gateway Road Event with intent to receive both incidents and scheduled events.
- Filtering – which allows the subscriber to specify the methods to filter the data publications of a subscribed type, i.e. the Gateway Road Event subscriber is able to receive data originating from only a specified list of data sources.
- State query – which allows the subscriber, upon subscription, to receive the current active state of the Gateway. This service significantly enhances the typical P/S implementation that deals only with real time publication.
- Quality of Service – which allows the subscriber to choose a desired reliability of message delivery from Unreliable, Best Effort, and Guaranteed. Initially, the Gateway will support Best Effort quality of service, i.e. the Gateway will retransmit the message a configurable number of times if the message has not been properly delivered.

DATABASE MANAGEMENT SYSTEM SELECTION
The database management system (DBMS) is the core of the Gateway system. The Gateway DBMS provides permanent data storage, transaction integrity, data integrity, rollback capability, and continual operations. While the adoption of CORBA and use of Java and C++ as the primary programming languages determined that a significant portion of the Gateway would be object oriented, a decision as to whether an object oriented DBMS (OODBMS) or a relational DBMS (RDBMS) would be used for the Gateway DBMS was also required.

A RDBMS stores the data in two-dimensional tables, each consisting of rows and columns. An RDBMS supports basic data types such integers and characters, and date time constructs. Complex data structures have to be broken down into individual simple data items and their relationships represented by relationships between tables for storage in an RDBMS. This means that two separate data models have to be maintained one for data storage and one for
in-memory manipulation. Conversions of data between the in-memory data model and the storage data model must be performed whenever a data item is exchanged between the database and its client program. Such conversions may significantly degrade the system run time performance. Two separate programming languages are typically required for regular in-memory data manipulation and database manipulation.

An OODBMS employs the same data model for both in-memory data manipulation and storage. The OODBMS preserves the object encapsulation, inheritance, polymorphism and other object orientation features.

The major database selection criteria for the Gateway database decision included: performance, price, reliability, ease of use, and maintenance requirements. The selection process boiled down to quantitatively comparing the run time performance difference between an OODBMS and RDBMS. Although benchmarking results were available in the public domain, they were inconsistent and inconclusive. Therefore, a series of benchmarking tests representative of typical Gateway database operations were conducted for the Gateway database design.

One leading RDBMS and one leading OODBMS product were selected for the testing. The initial test programs were developed in C++ for both the OODBMS and RDBMS. In order to compare the performance difference between a C++ and Java implementations, the test program for OODBMS was re-written in Java. All test cases were run on the same or comparable platforms.

The OODBMS demonstrated significantly superior run-time performance. This improved performance outweighed the moderately higher purchasing cost and led to the selection of an OODBMS for the Gateway TIS. In addition, the relatively slow performance of the DBMS Java interface, which is due to the fact Java programs are not fully compiled and need to be interpreted at run time, led to the decision that the CORBA layer for DBMS should be implemented in C++ in the Gateway TIS.

GATEWAY WEB SITE
The Gateway web site consists of a number of graphical and textual pages designed to provide user customizable content rich in dynamic information. These pages are organized in a way that is easy for user to access and navigate. Figure 3 is an overview of the web site organization. In addition to a set of public pages, the Gateway web site also serves over https protocol a set of password protected web pages for authorized users.

Text Report Pages
In essence, all types of information collected by the Gateway are made available to web users as a web reports, which include:

- Travel Time Report
- Congestion Report
- Incident Report
- Construction Report
- Special Event Report
- Dynamic Message Sign Report
- Weather Sensor Data Report
- Highway Advisory Radio Text Report

As the GCM corridor is a relatively large area, two mechanisms are available for users to acquire customized web report. A MyTravel web page allows users to select locations where
information is needed, and this selection can be saved as cookies (if allowed by the users). The Gateway web site will then make use of this location selection to filter out web information that is irrelevant to a user when a request for reports from this user. In addition, a user can select from a list of locations when requesting a report.

Figure 4 is a screenshot of Gateway Travel Times Report.

Map Pages
Map pages allow an integrated visual display of information. Figure 5 is a screenshot of one Gateway map page. A Gateway map page contains two maps:
- Main Map – used mainly for information display purpose
- Mini Map - used mainly for easy navigation across the corridor

The Gateway map provides a wide range of map operations rarely available in other online traffic information map pages. Below is a list of major Gateway web map features:
- Layered Map Structure - The Main Map is structured into a number of different layers for each major type of information display. The user is capable of selecting layers to be displayed, and the user selection of layered can be remembered if cookies are allowed by the users.
- Point to Pan – User can click at the Mini Map to cause the Main Map re-centered at the clicked point.
- Zoom In and Out – User can perform zoom-in and zoom-out functions by clicking on icons.
- 4-Way Scroll – User can click the border of the Main Map to scroll
- Click for Detail – The Main Map contains an assortment of clickable icons that will provide user detailed information on certain map features, including: travel time, congestion level, incident, construction, road, and jurisdiction.
- ToolTip – This feature provides users quick help where users hesitate.
- Linear Location Representation – Locations of many traffic related events have significant dimension longitudinally along the road, such as lane closure on a road between two cross streets far apart. Other traffic web maps due to the complexity involved often ignore this longitudinal dimension. The Gateway web map is designed to accurately render locations where longitudinal dimension is significant.

CONCLUSION
Developing a large scale ITS system like the Gateway based on a new technology like CORBA has proven to be very challenging. But the solutions to the integration and interoperation of a wide variety of heterogeneous systems in a large geographic region made possible through the CORBA-based design has also proven to be truly rewarding. Additional information about the Gateway TIS and its related ITS projects can be found at the GCM ITS Priority Corridor website [7].

Systems that adopt NTCIP CORBA C2C currently have no appropriate standards to follow in terms of data format and system interface. Development of CORBA IDL based national ITS center system interface specifications is needed to achieve the levels of interoperability envisioned in the National Architecture for ITS. The data models developed for the Gateway TIS for location referencing, roadway device, and event and traffic data exchange are offered as a starting point for national ITS standards and as a future direction for transportation computing.
REFERENCES
[1] GCM Public Information Center Library Website: www.gcmpic.ai.uic.edu/piclib
[3] NTCIP Website: www.ntcip.org
[5] ITE Website: www.ite.org
[7] GCM ITS Priority Corridor Website: www.travelinfo.org
LIST OF FIGURES
Figure 1: GCM Corridor Multi-Modal Traveler Information System Architecture
Figure 2: Functional Overview of Gateway TIS Architecture
Figure 3: Gateway Web Site Organization
Figure 4: Gateway Travel Times Report
Figure 5: Gateway Web Map
Figure 1 GCM Corridor Multi-Modal Traveler Information System Architecture
Figure 2 Gateway Functional Architecture

Functional View of System Architecture
Figure 3. Gateway Site Map

Figure 4 Gateway Travel Times Report
Figure 5 Gateway Web Map