

Exploring Data Transfer Methods in an ITS Environment

Jeremey Nunnally

(www.cbu.edu/~jnunnall)

Research Alliance for Minorities Program (RAM)

Christian Brothers University

August 15, 2001

Acknowledgements

Special thanks to ORNL and RAM for allowing me to research XML and CORBA, mentors Rekha Pillai, Robert Gibson, Charlie Davis, and Cheng Liu for their assistance and insight on all research performed on this project. The author would also like to thank partner Janine Lafayette for assistance in research on XML technology.

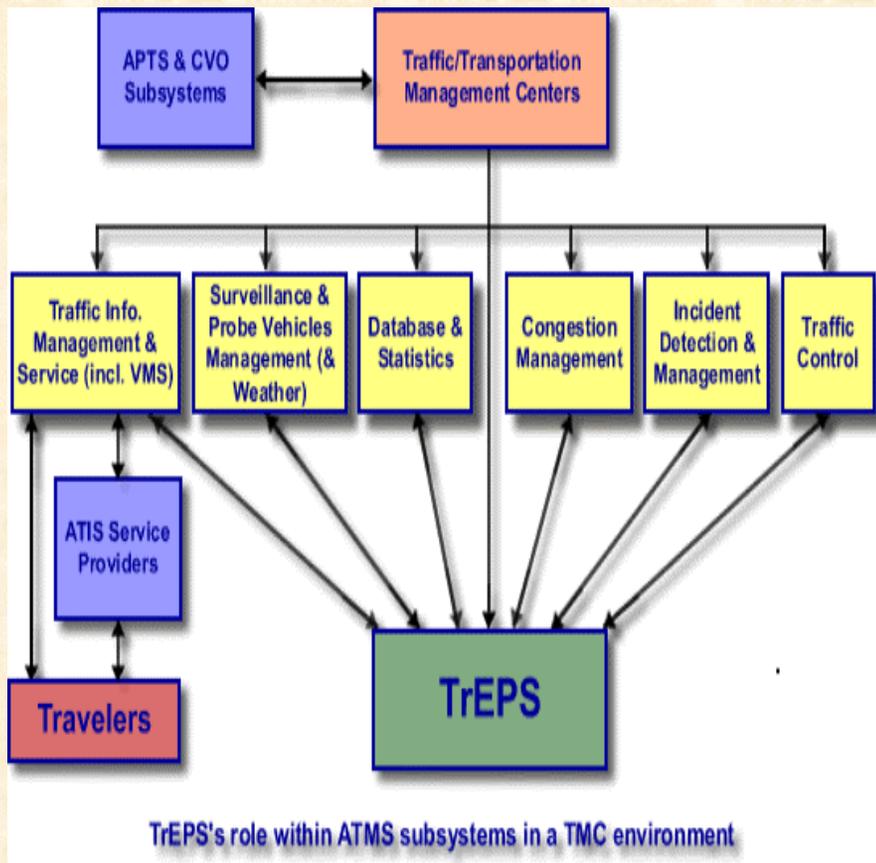
Overview

- ❑ Intelligent Transportation Systems
- ❑ XML
- ❑ CORBA
- ❑ CORBA vs. XML
- ❑ Conclusions and areas for further research

Intelligent Transportation Systems

- ❑ Merge people and transportation with existing and emerging traffic control and information technologies
- ❑ ITS goals
 - **Safety**
 - **Efficiency**
 - **Productivity**

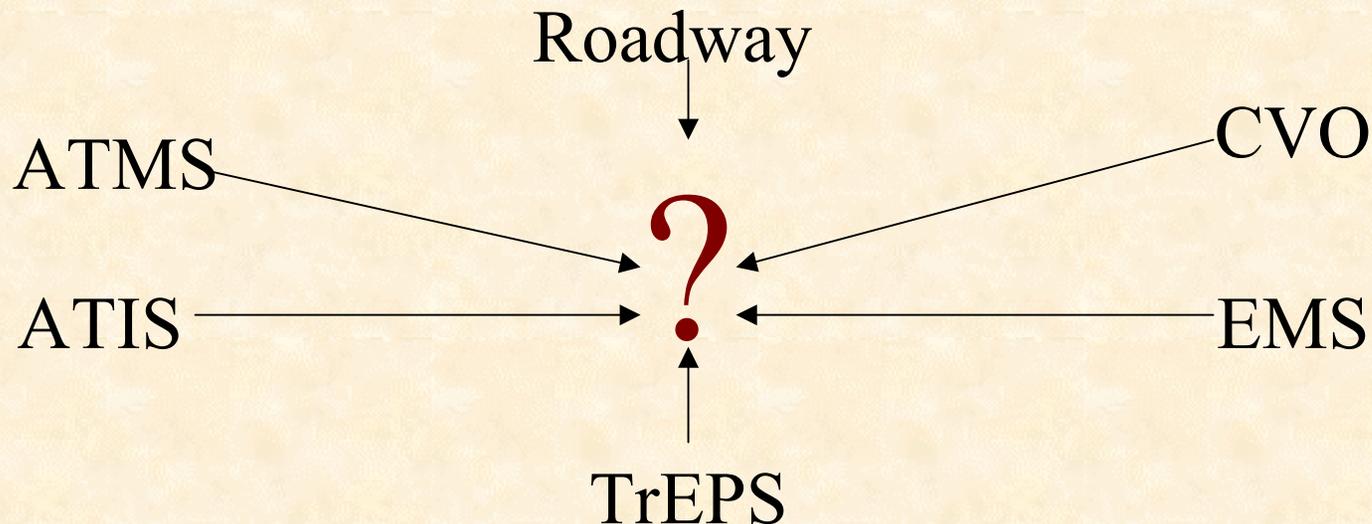
ITS Subsystems



- ❑ Advanced Traffic Management Systems
- ❑ Advanced Traffic Information Systems
- ❑ Roadway Systems
- ❑ Commercial Vehicle Operations
- ❑ Emergency Management Systems

The Problem

Enabling platform-neutral data transfer in a large, diverse, dynamic, data-rich ITS environment



Enter XML

- ❑ The Extensible Markup Language (XML) is a specification of the World Wide Web Consortium (W3C)
- ❑ Essentially, it is a standardized (XML 1.0) way of formatting data as a text file called an XML document
- ❑ Like HTML except tags mean whatever you want them to

XML is actually a set of specifications

- ❑ XML 1.0 defines syntax of an instance document
- ❑ XPath allows for the addressing of parts of an XML document
- ❑ XLink allows elements to be inserted into XML documents in order to create and describe links between resources
- ❑ XQuery allows for the querying of XML data sources

XML Strengths

- ❑ XML is not subject to any proprietary licensing
- ❑ Platform-independent
- ❑ Easy to learn and use
- ❑ Ability to actually store data
- ❑ Easy transfer over the Web

Weakness

As a text file an XML document must be parsed into some machine-readable. Parsing can be an expensive process in terms of both time and computing resources, especially for large documents (large datasets)

Possible Application in ITS

- ❑ Storage of surveillance and scenario data
- ❑ A standard data format to provide to many different users for unknown applications
- ❑ Defining a new traffic specific markup language

CORBA

- ❑ The Common Object Request Broker Architecture (CORBA) is a specification of the Object Management Group (OMG)
- ❑ CORBA was designed to handle distributed data and functionality encapsulated as single object
- ❑ Critical components
 - **Interface Definition Language (IDL)**
 - **Object Request Brokers (ORB)**

The Interface Definition Language (IDL)

- ❑ The CORBA IDL is a language-independent way to define object interfaces (data and procedures)
- ❑ A language-specific IDL compiler generates stubs and skeletons based on the IDL definition of an object

The Object Request Broker (ORB)

The ORB performs a number of critical functions, but one of its key functions is to coordinate stub-skeleton transactions - keeping track of the location and state of each interface

IDL and ORB Work Together

In a simple request, the client makes a method request to the ORB through an IDL stub. The ORB conveys the request to the object implementation via the IDL skeleton. Any result is returned along the same path.

CORBA Strengths

- ❑ The flexibility of the CORBA specification allows for a wide array of implementations.
- ❑ CORBA is faster than XML
- ❑ CORBA is a complete solution
- ❑ CORBA can protect sensitive data through object encapsulation and other specialized security measures

CORBA Weaknesses

- ❑ The cost of implementing a CORBA system may vary greatly
- ❑ CORBA systems are difficult to set up and may require a significant amount of time to implement especially on server end
- ❑ Where only data is needed (no methods) CORBA may very well be overkill

Possible Application in ITS

- ❑ Interfacing with field devices
- ❑ Handling sensitive data in a distributed system
- ❑ CORBA technology may also prove useful in Traffic Management Centers where Graphical User Interfaces (GUIs) must generate images based on data from various sources in different formats

XML vs. CORBA

XML

- Data Only
- Allows Data Persistence
- Easy to Learn and Use
- Easy transfer over the Web
- Requires parsing
- Cost-effective

CORBA

- Data and Functionality
- Data must be handle immediately
- Difficult to learn and use
- Uses special object protocols
- Pre-compiled IDLs
- Variable cost

Conclusion

- ❑ XML and CORBA both offer strengths that are applicable in ITS
- ❑ They are, in fact, closer to complementary technologies rather than competitors

Further Topics for Additional Research

- ❑ Alternatives to XML and/or CORBA
- ❑ Quantitative performance analysis of XML and CORBA in ITS scenarios
- ❑ Economic factors of implementing new technologies in ITS