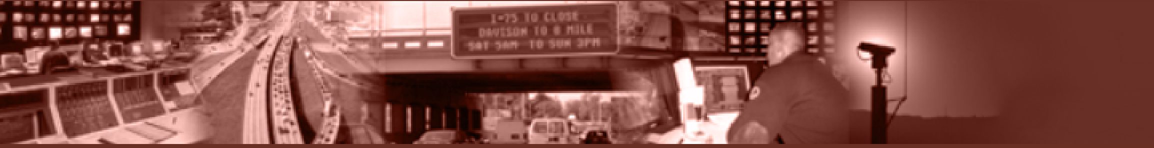


Decision Support for Traffic Management Systems—Current Practices

PUBLICATION NO. FHWA-HRT-21-108 NOVEMBER 2021

U.S. Department of Transportation
Federal Highway Administration

Research, Development, and Technology
Turner-Fairbank Highway Research Center
6300 Georgetown Pike
McLean, VA 22101-2296

FOREWORD

New and emerging sources of data generated from travelers using mobile devices, vehicles, infrastructure, and other sources will provide agencies with opportunities to change how they manage traffic and their transportation systems. These sources of data offer agencies with traffic management systems (TMSs) the challenge of enhancing or developing capabilities to collect, compile, save, use, and share these data.

The Federal Highway Administration Office of Operations Research and Development is pleased to present *Decision Support for Traffic Management Systems—Current Practices*. This report examines current practices among agencies using or planning to implement decision-support tools (DSTs) to improve the capabilities and performance of their TMSs. The report also provides a framework for understanding types of available DSTs, how they may enhance decisionmaking and improve TMS management and operation or the use of specific operational strategies, and how DSTs can be integrated into a TMS to support or carry out real-time decisionmaking.

Brian P. Cronin, P.E.
Director, FHWA Office of Safety and Operations
Research and Development

Recommended citation: Federal Highway Administration, *Decision Support for Traffic Management Systems—Current Practices* (Washington, DC: 2021) https://doi.org/10.21949/1521707.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA-HRT-21-108		

4. Title and Subtitle Decision Support for Traffic Management Systems—Current Practices	5. Report Date November 2021
	6. Performing Organization Code: HRDO-20
7. Author(s) Kevin Miller, John Horner, Jeff Adler, Robert Sanchez	8. Performing Organization Report No.
9. Performing Organization Name and Address Leidos, Inc. 11251 Roger Bacon Drive Reston, VA 20190 Kapsch TrafficCom USA, Inc. 8201 Greensboro Drive McLean, VA 22102	10. Work Unit No. HRDO-20
	11. Contract or Grant No. DTFH61-16-D00053
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Highway Administration 1200 New Jersey Avenue, SE Washington, DC 20590	13. Final Report; September 2017–November 2021
	14. Sponsoring Agency Code HRDO-20

15. Supplementary Notes
Task order manager: Jon Obenberger (HRDO-20; ORCID: 0000-0001-9307-847X)

16. Abstract
Traffic management systems (TMSs) and their operating centers, traffic management centers, support real-time and offline functions, services, actions, and activities—as well as related tasks and decisions (e.g., administrative actions and condition reports). In this context, decision-support tools (DSTs) are not generally used as standalone solutions; rather, they are deployed and integrated in tandem with one or more specific TMS subsystems (e.g., software subsystems and data subsystems). A DST can be integrated into a TMS subsystem's software, component, or device—depending on a TMS's requirements and uses for the DST. This report examines current practices and trends among agencies using or planning to implement DSTs to improve TMS capability and performance. It provides a framework for understanding types of DSTs available and how they may support and carry out decisionmaking within a TMS. It also discusses potential benefits of DSTs, issues to consider in integrating various DSTs into the real-time operation of a TMS, and current practices to consider in planning and developing next generation TMSs.

17. Key Words	18. Distribution Statement
Decision support, decision-support tools, traffic management system, traffic management center	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161. http://www.ntis.gov

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	96	N/A

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized.

SI* (MODERN METRIC) CONVERSION FACTORS				
APPROXIMATE CONVERSIONS TO SI UNITS				
Symbol	**When You Know**	**Multiply By**	**To Find**	**Symbol**
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1,000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2,000 lb)	0.907	megagrams (or “metric ton”)	Mg (or “t”)
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa
APPROXIMATE CONVERSIONS FROM SI UNITS				
Symbol	**When You Know**	**Multiply By**	**To Find**	**Symbol**
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or “t”)	megagrams (or “metric ton”)	1.103	short tons (2,000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	2.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380.
(Revised March 2003)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	application programming interface
CCTV	closed-circuit television
CHART	Coordinated Highways Action Response Team
ConOps	concept of operations
COTS	commercial off-the-shelf
DMS	dynamic message sign
DSS	decision-support system
DST	decision-support tool
ERE	expert rules engine
ETL	extract, transform, and load
FDOT	Florida Department of Transportation
FHWA	Federal Highway Administration
GUI	graphical user interface
HAR	highway advisory radio
HCM	Highway Capacity Manual
HCS	Highway Capacity Software
ICM	integrated corridor management
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
ISO	International Organization for Standardization
ITIL	Information Technology Infrastructure Library
IT	information technology
ITS	intelligent transportation system
LOS	level of service
NOAA	National Oceanic and Atmospheric Administration
TMC	traffic management center
TMS	traffic management system
TRB	Transportation Research Board
UI	user interface

CHAPTER 1. INTRODUCTION

Traffic management systems (TMSs) are complex operational systems that combine field equipment, advanced communications, information technology (IT), and software tools. TMSs collect and synthesize traffic data, integrate external systems, and enable the command and control of intelligent transportation system (ITS) field devices. TMSs help operators actively manage and perform a range of functions and actions to facilitate improvements in the efficiency, safety, and predictability of travel on the surface transportation system.

Agencies today have an expanded amount of traffic data available to them and a growing range of operational strategies; which increase the complexity of the real-time management and operation of TMSs. Decision-support tools (DSTs) encompass the universe of decisionmaking support, including computer-based and noncomputer-based. Examples of offline DSTs include decision trees and decision tables that can be printed and collated into operator or TMS operations reference guides. Computer-based DSTs can potentially play a vital role in improving traffic operations personnel's real-time decisionmaking, thereby complementing and enhancing the operational capabilities and performances of TMSs.

DSTs, as described in this report, are computer-based tools that can process vast amounts of data, capture the operational processes of an organization, and potentially mimic the real-time decisionmaking of human operators of TMSs. DSTs can aid operations personnel in monitoring and assessing network conditions, detecting and verifying adverse conditions, and identifying and evaluating appropriate response strategies to planned and unplanned events. DSTs can also help agencies achieve more consistent and understandable decisionmaking across their transportation management staff.

This report examines current practices and trends among agencies using or planning to implement DSTs to improve the capabilities and performance of their staff and TMSs. This report also provides a framework for understanding the types of available DSTs, how these tools may benefit operations personnel, and how DSTs can be integrated into a TMS to support or carry out real-time decisionmaking.

The primary objectives of this report are as follows:

- To provide readers with an understanding of the types of DSTs available to assist them in improving the management and operation of TMSs.
- To identify issues for agencies to consider when assessing decision-support needs and TMS capabilities.
- To frame issues for agencies to consider when integrating DSTs into the management and operation of TMSs.
- To identify available opportunities during the implementation of DSTs in any phase of the TMS lifecycle (e.g., planning, design, maintenance, and operation).

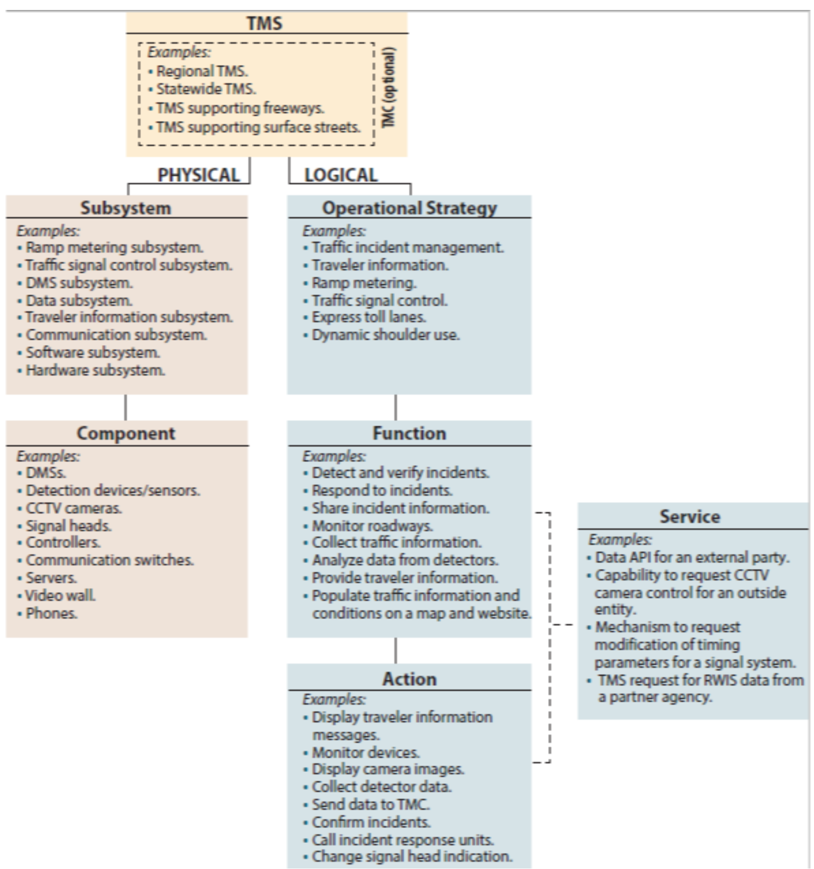
The intended audience for this report is public agency staff, academia, and contractors involved in planning, designing, implementing, maintaining, managing, and operating TMSs. This report should allow readers to gain further appreciation for various types of tools with the potential to improve the decisionmaking capabilities and performance of a TMS. Readers are assumed to have a general awareness of ITS technologies or TMSs.

The remainder of this chapter provides an overview of DSTs and presents key issues for agencies to consider when assessing current capabilities of TMSs. Additionally, it outlines considerations for integrating decisionmaking tools into real-time TMS operation and DSTs into TMS planning and design.

DECISIONMAKING AND TRAFFIC MANAGEMENT SYSTEMS

Improving the timeliness and effectiveness of real-time decisionmaking is a central focus for agencies that manage and operate transportation networks. Traffic management requires real-time decisionmaking for monitoring traffic conditions, detecting and managing unplanned events, scheduling and managing planned events, improving signal timing, assessing and planning for adverse weather, and managing critical infrastructure. Good decisionmaking requires a skillset that includes strong knowledge of a transportation system, a clear understanding of an organization's operational procedures and their real-time application, and the processing and assimilation of a wide range of data and information.

TMSs provide the software platforms and subsystems that agencies and traffic management centers (TMCs) rely on to support their operations. Figure 1 illustrates the physical and logical components of a TMS along with a series of examples. Operational decisionmaking is associated with the logical elements of the TMS structure—the operational strategies of the organization that the TMS is designed to support, the functions supported by the TMS, and the actions to be executed.



API = application programming interface; CCTV = closed-circuit television; DMS = dynamic message sign; RWIS = road weather information system.

Figure 1. Diagram. Traffic management system structure with examples.

The decisionmaking framework associated with traffic management is shown in figure 2 and involves four decision stages, which have the following functions:

- Monitor: Collect and process data from various field devices, third-party data sources, and partners to evaluate current conditions in the transportation network.
- Calculate and predict: Apply advanced data processing that combines current information with historical information to predict the future state of the network and any increased risk of impactful events. This phase also involves detecting and predicting events that will adversely impact traffic performance and warrant an operational response.
- Propose: Generate one or more response plans, such as sets of operational strategies, functions, and actions, to mitigate the effects of traffic events.
- Select and implement: Select and execute the response plan deemed most likely to most effectively improve performance.

Early generations of TMSs typically relied on operations personnel to manually perform activities associated with the four decision stages. Today, available computer-based DSTs can reduce this burden on operations personnel. DSTs can facilitate data processing and automate forecasting and prediction. They can provide visualization tools that improve a TMS's monitoring of current conditions and prediction of future conditions. They can also make recommendations to maximize operational strategies, functions, and actions. As such, they can be integrated with TMSs to fully or partially automate processing and decisionmaking activities across these four decision stages.

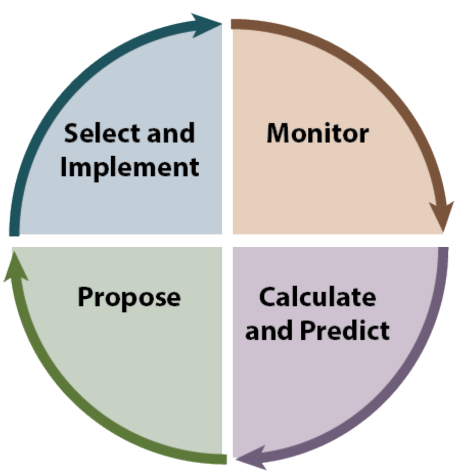
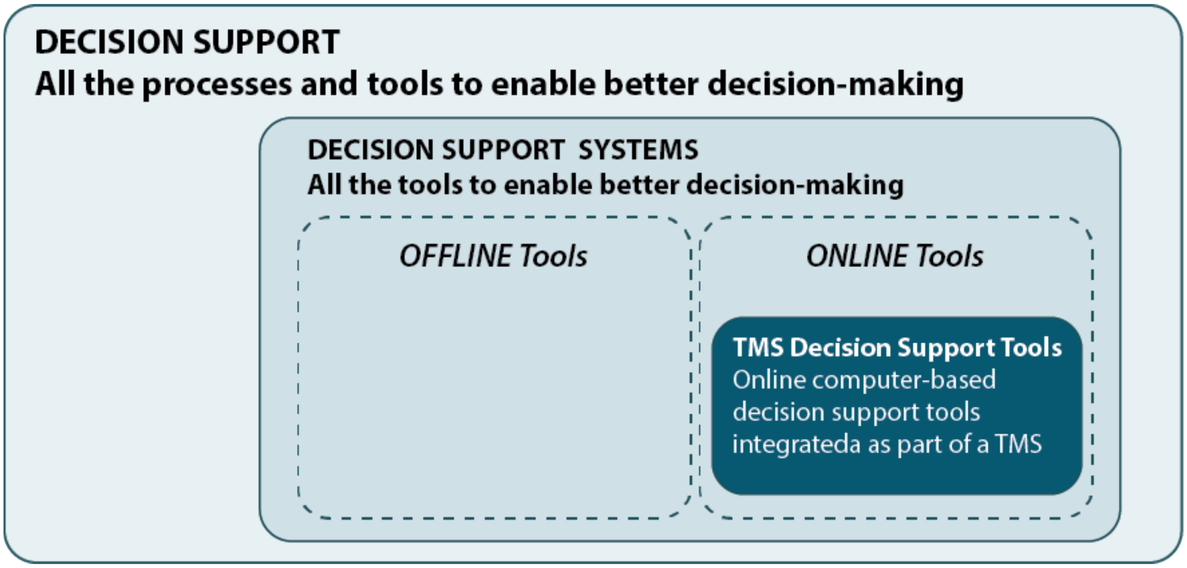


Figure 2. Diagram. Traffic management decisionmaking.

WHAT IS A DECISION-SUPPORT SYSTEM?

Decision support is an overarching concept that embraces all the processes and tools that enable better, faster, and more consistent decisionmaking. Computer-based decision-support systems (DSSs) have entered various industries and are used all over the world to improve business operations. In this report, the following related terms are used but have distinct meanings: *DST* and *DSS*. Figure 3 depicts the taxonomy of DSTs as adopted for this report, which focuses on the category of TMS DSTs seen in the diagram.



Source: USDOT.

Figure 3. Illustration. Taxonomy of decision support.

Decision-Support Tools

DSTs encompass the universe of decision support, including computer-based and noncomputer-based. Examples of noncomputer-based include paper-based decision trees and decision tables that can be printed and collated into a reference guide.

Decision-Support Systems

DSSs are computer-based tools that support business or operational decisionmaking activities. This term encompasses a wide range of technologies and solutions used for online or offline decisionmaking, as follows:

Offline DSSs: Off-line systems that are typically used to support short- to long-term planning. Maintenance DSSs and pavement management DSSs are examples of offline systems that agencies commonly use to support maintenance activities.

Online DSSs: Real-time, computer-based DSSs that are well suited to support real-time traffic management and operational decisionmaking. This report primarily focuses on these types of DSSs.

Examples of offline DSTs for TMSs include the following:

- Incident response plans: Provide instructions for pre-established responses (e.g., plans and procedures) to various types of incidents, including processes for coordinating efforts among various agencies and service providers. Agencies typically have plans developed to respond to a wide range of potential incidents, wherein the actions taken vary based on location, time of day, severity, and conditions (e.g., traffic demand and weather). These plans enable consistent, rapid decisions and define how agencies and service providers respond to various types of incidents without needing to reinvent the wheel each time an incident arises.

- Decision trees: Utilize tools that allow human operators to complete structured processes to reach decisions that are consistent with their organization's accepted policies and procedures. Through a series of simple questions (usually with a few limited answers), users can arrive at clear decisions regarding specific issues.

- Performance monitoring, measuring, and reporting tools, including spreadsheets and algorithms: Generate individual measures of performance or summaries of overall performance based on current observed conditions and historical data. Examples of individual measures include travel time indices and travel delay data. Examples of summaries include dashboards and reports. These summaries are typically generated by TMSs in real-time, daily, monthly, or annually and provide a snapshot of how an operation is performing based on industry-recognized measures. Such information enables policy makers and agency leaders to make better long-term decisions. It also meets a need for the general public to better understand how the system is performing.

Following are examples of online DSTs for TMSs; these examples include tools that have been integrated into and support real-time TMS management and operation.

- Real-time traffic analysis tools: Have the potential to analyze data (e.g., traffic conditions, weather, and transit system performance) and can be integrated into the decisionmaking of how a TMS is managed and operated. For example, these tools can analyze data to enable improved decisionmaking for traffic signal timing.

- Look-up tables for traffic operational strategies and control plans: Simplify the process of selecting operational strategies (e.g., ramp metering and lane control), control plans (e.g., time of day or demand responsive, open or closed), and actions to be implemented (e.g., incident response plans) based on conditions detected or projected to occur. Although look-up tables of incident response plans may be offline tools, they enable a simplified, real-time process for selecting the appropriate plan(s).

ASSESSING, PLANNING, AND DESIGNING INTEGRATION OF DECISION-SUPPORT TOOLS

As agencies continue to plan for next-generation TMSs, they are considering what enhancements may be needed and what may be reasonable and feasible to implement to improve TMS performance, decisionmaking, and automation. As many existing subsystems and components can be modified to support TMS integration, agencies should evaluate whether their current TMSs can be modified to enhance functionality. Based on this evaluation, an agency can decide whether it will reuse and modify an existing TMS or design and build or buy a new one. Such evaluations should examine the following questions:

- What technologies may be appropriate to select.
- What may be required to support the integration and use of these new technologies.
- What data may be needed for various functions and services.
- What translations (e.g., format changes) may be needed to enable data to be used for various functions or services.
- What tools and software should be stand-alone and what tools and software should be integrated into the software subsystem of the TMS.
- What, if any, commercial off-the-shelf (COTS) or open-source software products are acceptable or what commercial or proprietary product would be more appropriate.

Once the requirements and capabilities of a current TMS have been established and compared to desired requirements and capabilities, an agency should evaluate available products and technologies. If the agency concludes that no COTS products or internal reusable product components are suitable, then the agency should consider the option of building their own TMS. Staff should identify and assess the potential build/buy/reuse decision alternatives (e.g., COTS, open-source, or proprietary) for each component of the TMS that may need to be revised. After analysis is conducted and decisions are reached, staff should document recommended selections for technologies and products for each component. This process should be considered part of the system design.

These issues and technologies are just some of the factors that agencies need to consider. These decisions are further discussed in chapter 5, which also includes more details on the specific evaluations and decisions that agencies have made when integrating DSTs into their TMSs.

OVERVIEW OF REPORT

This report provides an overview of current practices among agencies integrating DSTs into the real-time management and operation of a TMS. It also discusses issues to consider when integrating DSTs into existing or new systems, requirements to consider for DSTs and TMSs, and tool integration details to consider.

The study team collected the information in this report by conducting a literature review on existing research, TMS deployments, and DSTs used in the transportation industry. Additional sources of information included material available from resources for transportation agencies, including the Federal Highway Administration (FHWA) ITS Program. The study team also reviewed information provided by individuals and agencies during the 2018 Transportation Research Board (TRB) Workshop on DSTs for the Next-Generation of TMS, information from sessions and meetings of the TRB ACP20 Freeway Operations Committee, and information from National Cooperative Highway Research Program Reports. The remaining chapters in this report, which synthesize the study team's analysis of current practices in integrating DSTs and operating TMSs with its research, are summarized in table 1.

Table 1. Overview of chapters 2–7.

Chapter	Title	Description
2	Decisionmaking and DSTs	Discusses the uses of various DSTs in assisting agencies in managing and controlling traffic.
3	Integrating DSTs into TMSs	Summarizes the basis, needs, benefits, implications, and options for integrating DSTs into TMSs.
4	TMS Decision-Support Needs and Requirements	Summarizes the needs and implications involved in integrating DSTs into TMSs.
5	Designing, Developing, and Deploying DSTs in TMSs	Summarizes key issues to consider when integrating DSTs into the planning, design, development, and operation of TMSs.
6	Monitoring, Evaluating, and Updating DSTs in TMSs	Presents strategies to evaluate DST performance and determine when changes may be needed regarding specific TMS and DST functions, tasks, and capabilities.
7	Examples of DSTs for TMSs	Provides case studies of TMSs with DSTs for the three types of DSTs described in the report.

CHAPTER 2. DECISIONMAKING AND DECISION-SUPPORT TOOLS

This chapter provides an overview of the decisionmaking processes and DSTs agencies use to sustain traffic operations programs, plan for day-to-day traffic operations, plan to improve TMSs, and operate TMSs and systems with a TMC, also known as an operations center. This chapter summarizes key factors influencing decisionmaking, describes the concept of decision support, and highlights the typical types of DSTs used by agencies. It also examines how DSTs are being used both offline and online and how agencies' operational policies and procedures may affect the design and use of a DST by a TMS.

This report focuses on the use of online DSTs to support current methods of TMS management and operation. However, understanding more general concepts associated with decision support and the range of tools available to agencies provides the reader with a better understanding of the potential benefits these tools may offer for the next generation of TMSs. The objectives of this chapter are to address the following points in detail:

- Discuss what factors influence decisionmaking.
- Describe how an organization's operations-related policies and procedures affect its decisions for designing and operating a TMS.
- Describe how various types of DSTs can be used for various types of decisions, tasks, and functions carried out within a TMS.
- Present examples of how offline and online DSTs can be used within TMSs.

This chapter helps the reader understand the concept of decision support, tools used to assist in operations-related decisionmaking, types of tools that may be appropriate to integrate into the day-to-day operation of a TMS, and how these tools are integrated and used by TMSs. Following this chapter, chapter 3 identifies how these tools and processes have been integrated and used to automate or improve specific TMS tasks, functions, and operations.

TRAFFIC MANAGEMENT SYSTEMS

TMSs are the central tool that agencies use to improve the management and operation of the surface transportation system. TMSs also share information with and support other traffic systems management and operations programs or initiatives. TMSs provide the functions, services, and capabilities that agencies need to manage traffic, control ITS field devices, manage incidents and planned events, coordinate with stakeholders in response to these events, and provide information on travel conditions.

A key feature of TMSs is that they provide decision support to help operators make faster and more effective decisions in response to current and projected conditions (e.g., traffic, roadway infrastructure, weather, and events).

Decision-support capabilities have expanded to support or automate functions, decisions, or actions that TMS operators carry out or implement (e.g., ramp metering and identifying travel times). Recently designed TMSs can provide responses to multiple events. With increased focus on integrated corridor management (ICM), the newer TMSs have also begun the transition from managing, controlling, and coordinating travel on one type of facility to supporting several facilities (e.g., freeways, surface streets, and traffic signals).

FACTORS INFLUENCING DECISIONMAKING

Decisionmaking is often an inherently complex process, and the challenges it presents may be compounded by additional factors, including the experience level of decisionmakers and other factors such as societal pressures, cultural norms, fatigue, inherent biases, and other limitations. These factors' effects on decisionmakers can impact the quality of policies, procedures, programs, actions, services, and operations that result from their decisions. The effects of these limitations and biases can become even more pronounced as systems grow in complexity, when decisions need to be made quickly, or when there is insufficient data or time to assess an incomprehensible amount of data. DSTs are designed to address these limitations, and over the years, these tools and associated processes have demonstrated their effectiveness across a variety of fields (e.g., medical diagnoses and emergency response).

As transportation networks have grown more complex and more congested, decision support has played a greater role in optimizing the efficiency and safety of the movement of people and goods. With the amount of information available for these networks continuing to grow, DSTs can help process data and improve the quality of the decisions that are made, which will influence the performance of these complex systems. The use of established processes and logical aids can offset some of the typical biases that arise when making decisions. Robinson et al. (2017) found that DSTs can minimize the influence of biases (e.g., confirmation bias) on decisions, ensure consideration of alternatives, and encourage quantitative assessment of options.

However, the same study also found several potential disadvantages to an over-reliance on DSTs, such as discounting intuition and experience. Despite these pitfalls, properly designed and implemented DSTs can be valuable assets to human operators of TMSs. In particular, Lukasik et al. (2011) found that the use of DSTs in transportation can leverage and improve on a number of recent advances in real-time traffic management and operation.

The Concept of Decision Support

The concept of decision support emerged primarily from the theoretical studies of organizational decisionmaking that began at the Carnegie Institute of Technology (Simon 1955; Cyert and March 1963) in the late 1950s and the implementation work that followed in the 1960s. This initial work generally focused on higher-level, strategic decisions, but the field quickly expanded to include tools for supporting all types of decisions. Decision support became a distinct area of research in the mid-1970s and continued to gain in intensity during the 1980s.

In the mid- to late-1980s, the following three key forms of computer-based DSSs emerged, accompanied by an evolution from single-user to multiple-user approaches.

- Executive information systems: Programs that ran on mainframe computers to package a company's data and provide sales performance or market research statistics for decisionmakers. The intent was to develop computer applications that highlighted information to satisfy the company's strategic and program planning needs. Typically, an executive information system would only provide high-level data that supported these decisions.

- Group DSSs: Computer-based systems that evolved from single-user to multiple-user. These systems were designed to use inputs from numerous users interacting simultaneously with the system to arrive at a decision as a group.

- Organizational DSSs: Computer-based systems that focused on the coordination and dissemination of decisionmaking across functional areas and hierarchical layers. A key goal was to ensure that decisions were congruent with the organization's goals and management's shared interpretation of the competitive environment.

Decision support continued to evolve as a field of study and practice during the 1980s. Since then, as computing hardware has gotten more powerful and as more data have become available, DSSs have continued to evolve and improve. Today, there are many commercial products available to build DSTs and provide decision-support functions across multiple industries. For example, the financial and medical industries use a variety of systems that evolved from the early research. These systems assist decisionmakers with various choices, including what to do throughout loan processes, which financial instruments to purchase, and which medicines to prescribe.

TYPES OF DECISION-SUPPORT TOOLS

DSTs used within a TMS can be classified on two levels—the approach of the tool and the type of interaction. Approaches of decision support tools are discussed in more detail in the next section. The type of interaction refers to the level of human interaction with the DST—how decisions are made (manually or automatically) and the degree to which the DST involves the decisionmaker (interactive or automated).

Common Approaches of Decision-Support Tools

The following sections outline the three common approaches of decision-support tools. Table 2 illustrates them.

Table 2. Decision-support tools mapped to decision-support classification.

Approach	Incident Response Plans (offline)	Decision Trees (offline)	Performance Measurement Tools (offline)	Real-Time Traffic Analysis Tools (online)	Look-Up Tables (online)
Knowledge-driven	Yes	Yes	No	Yes	Yes
Data-driven	No	Yes	Yes	Yes	Yes
Model-driven	No	No	Yes	Yes	No

Knowledge-Driven DSTs

Knowledge-driven DSTs provide specialized problem-solving expertise based on the processing of stored facts, rules, procedures, and other similar forms of knowledge. They attempt to emulate human reasoning but with more consistent results. "Expert systems" are the most well-known type of DSTs in this category; they use databases of knowledge generated by previous expert users, along with a set of system-specific business rules, to emulate the decisionmaking capabilities of an expert user. Based on this knowledge base, such tools recommend actions to human traffic operators. Knowledge-driven tools differ from table-based tools (e.g., decision tables, as described in chapter 1) in the way knowledge is extracted, processed, and presented. A knowledge-driven DST attempts to emulate human reasoning, whereas a table-based tool responds to all events in a predefined manner. Primary characteristics of knowledge-driven DSTs are as follows:

- They provide recommendations based on human knowledge.
- They apply heuristic (i.e., practical, rule-of-thumb) techniques for problem solving.

Data-Driven DSTs

Data-driven DSTs use data, typically from databases that are designed to be queried, to aid in the decisionmaking process—thus enabling the processing and analysis of data to develop insights that support decisionmaking. Statistical analysis software is one of the most common types of DSTs. The effectiveness of a data-driven DST depends on the quality of the data gathered and the effectiveness of the analysis and interpretation by the decisionmaker. With many of the data analysis tools currently being used by the transportation industry, agencies can customize dashboards to display the data they want to see and run custom reports. Ongoing advances in how data can be accessed, analyzed, and visualized are allowing an increasing number of agency staff without technology backgrounds to work with analytical tools, analyze data, and make better-informed decisions. Primary characteristics of data-driven DSTs are as follows:

- They summarize data into usable information.

- They use large amounts of data and employ well-organized methods to query and visualize the results of analysis they conduct.

- They offer flexible reporting and analytical capabilities.

Model-driven DSTs

Model-driven DSTs use mathematical models that express theoretical relationships among data elements or key variables of interest during analysis. These tools can be employed online or offline to simulate the behavior of a transportation system, or parts of it, using different values for certain parameters. They use a variety of analysis tools (e.g., statistical software and traffic analysis software) to assess available data, evaluate them, and report on conditions. Traffic analysis tools that use data captured by a TMS can be used online or offline to assess how a transportation network will perform based on various potential actions. Model-driven DSTs can be used in real time as a part of a TMS to predict possible outcomes of actions a TMS is considering implementing. These predictions allow agencies to assess impacts on key metrics such as travel times, environmental impacts, and person and vehicle throughputs. Primary characteristics of model-driven DSTs are as follows:

- They provide “what if” analysis based on historical and assumed (e.g., scenario-based) data.
- They leverage algorithms, simulations, and optimization tools to provide decision support.
- They use data and parameters provided by decisionmakers to help analyze a situation but do not require intense amounts of data input.

In summary, DSTs are computer-based tools that support business or operational decisionmaking activities. This term encompasses a wide range of technologies and solutions used for online and offline decisionmaking. Within the framework of these three approaches (knowledge-driven, data-driven, and model-driven), TMSs commonly use a variety of DSTs. These DSTs can be online or offline.

Types of Offline Decision-Support Tools

Offline DSTs are typically used to support short-term and long-term planning, programming, and policy-related transportation system management and operation activities. Maintenance DSSs and pavement management DSSs are examples of offline systems that agencies commonly use to support maintenance activities. The following sections provide examples of offline DSTs that are used in TMSs.

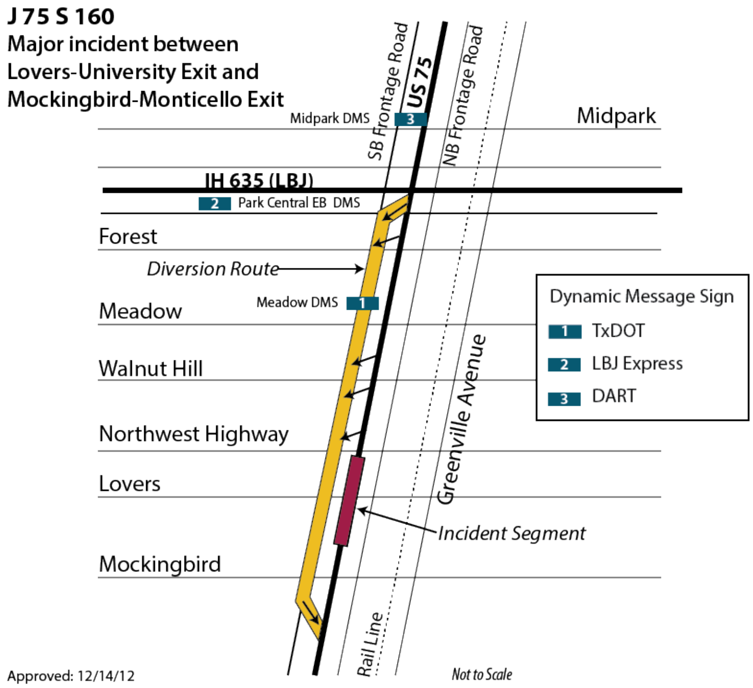
Incident Response Plans

Incident response plans are predefined sets of actions that agencies perform based on the location, type, and expected impact of an incident (e.g., disabled vehicle on shoulder or blocked lane or lanes); these plans can be online or offline. Within a TMS, the use of these plans as an offline tool could be as simple as an operator viewing a plan from a printed document and implementing the actions within a TMS. Many TMSs have developed incident response plan books that are based on experience. These plans provide operators with a playbook for how to respond to a particular incident, prescribing appropriate actions (e.g., temporary detours and traffic signal changes) and identifying necessary resources.

Offline incident response plans are usually selected using a simple process in which operators monitor TMS data and manually select a response plan based on variables such as incident location, time of day, level of congestion (e.g., length of queue), and availability of detour routes. (Some TMSs have converted these offline tools into online tools.) Offline incident response plans are static plans that are available from a document, either electronic or printed, which the operator uses to respond to an event.

Within some TMSs, software recommends to a human operator which incident response should be used based on business rules. Response plans may be based on the location, time of day, and severity of an incident. These plans are usually not integrated into the TMS, but a reference to the incident response plan is provided within the software. The TMS has basic business rules integrated into the software, which then select the appropriate incident response plan based on the rules.

Figure 4 illustrates a sample response for a major incident, including various TMS field devices and suggested locations to display messages on a dynamic message sign (DMS).



© 2010 Dallas Area Rapid Transit.
DART = Dallas Area Rapid Transit; EB= eastbound. LBJ Express = Lyndon Baines Johnson Expressway; NTTA = North Texas Tollway Authority; SB = southbound; TxDOT = Texas Department of Transportation; WB = westbound.

Figure 4. Diagram. Major incident response plan overview.

Based on the incident in the diagram, an incident response plan outlines an overall response and details all associated agency actions, as shown in figure 5.

Dallas

- Frontage road: activate signal timing signal plan #32 and monitor every 15 min.

Native Signal ID	Name	SmartNet Signal ID	DIRECT Signal ID
3455	CENTRAL @ UNIVERSITY	1010139110	2053
3449	CENTRAL @ MOCKINGBIRD	1010139040	2061
3453	CENTRAL @ SMU	1010139090	2081
3468	CENTRAL @ SOUTHWESTERN	1010139240	2112
3465	CENTRAL @ LOVERS	1010139210	2113
3473	CARUTH HAVEN @ CENTRAL	1010139290	2126
3496	CENTRAL SBFR @ WALNUT HILL	1010139520	2159
3491	CENTRAL SBFR @ PARK LN	1010139470	2160
3489	CENTRAL SBFR @ NORTHPARK S.C.	1010139450	2167
3501	CENTRAL SBFR @ MEADOW	1010139570	2178
5634	Central SB @ Churchill	1010142330	3141
3766	FOREST LN @ CENTRAL SBSR	1010142280	3166
3505	ROYAL LN @ CENTRAL SBSR	1010139610	583

- CCTV: monitor traffic conditions on Southbound US 75 Frontage Rd. every 15 min.

TxDOT

- DMS:

Message ID	Name	SmartNet ID
2	75_SB Meadow	5074420

LBJ Express

- DMS:

Message ID	Name	SmartNet ID
2	HOV 635_EB_Park Central	5074220

DART

- DMS:

Message ID	Name	SmartNet ID
2	HOV 75_SB Midpark	5074900

CCTV = closed-circuit television; DART = Dallas Area Rapid Transit; EB = eastbound; LBJ Express = Lyndon Baines Johnson Expressway; Rd = road; SB = southbound; TxDOT = Texas Department of Transportation; WB = westbound.

Figure 5. Screenshot. Agency actions in a response plan for a major incident.

Decision Trees

Decision trees utilize offline tools that allow human operators to complete structured processes to reach decisions that are consistent with their organization's accepted policies and procedures. Through a series of simple questions (usually with a few limited answers), users can arrive at clear decisions regarding specific issues.

Decision trees are typically used for planning, selecting additional equipment, or deploying specific technologies. Some decision trees have been developed to assist operators in consistently predicting incident durations. By providing a process with decision points and options, an operator works through each decision point to reach a conclusion or action at the end. The decision tree is basically a series of questions with potential answers that are linked together to create a decision once the questions have been answered.

For example, in the case of a Maryland Coordinated Highways Action Response Team (CHART) decision tree, when the TMS's operator learns of an incident, the DST asks the operator a series of questions to help determine an approximate incident duration. The DST then provides the operator with an expected duration, which is entered into traffic management software. Table 3 and table 4 show the variables and outputs of part of the decision tree and the relationships between incident clearance times and their associated factors. This information was derived from experience and historical data. Using a combination of the variables seen in these tables and the information from the questions asked of the operator, the decision tree provides the operator with an expected duration.

Table 3. Maryland Coordinated Highways Action Response Team incident duration decision tree.

Number	Description of Classifier			Clearance Time (minute)
1	IF	(road=I895 & incident_type=disabled) or (noTT=0 & noSDsh=0 & incident_type=disabled) or (noTT=0 & road=US50 & incident_type=disabled)	THEN	Minor (<30)
2	ELSE-IF	(OC=TOC3 & noLane=13 & county=MO & incident_type=cpd) or (noTT=0 & road=I495 & incident_type=disabled & pavement=dry) or (chart=1 & noLane=12 & road=I95 & incident_type=disabled)	THEN	Minor (<30)
3	ELSE-IF	(OC=TOC3 & SDBmain=minor & pavement=unspecified) or (OC=AOC_South & noLane=12 & road=US50) or (weekday & incident_type=disabled & detection=CHART)	THEN	Minor (<30)

Number	Description of Classifier			Clearance Time (minute)
4	ELSE-IF	(totalveh=2 & incident_type=fatality) or (night=0 & road=other & incident_type=fatality)	THEN	Major (>120)
5	ELSE-IF	(noTT=0 & county=3 & incident_type=disabled) or (OC=TOC3 & noSDBmain=0 & incident_type=cpd)	THEN	Minor (<30)
6	ELSE-IF	(noSUT=0 & nonholiday & exit=22 on I495, I270, I695, and US50) or (SDBmain=minor & county=MO & detection=CHART) or (noSDsh=2 & noSDBmain=0 & noODBsh=0 & incident_type=disabled)	THEN	Minor (<30)
7	ELSE-IF	(night=0 & noODBsh=0 & exit=31 on I495, I270, I695, and I83) or (noODmain=3 & SDBmain=minor & county=Anne Arundel) or (chart=1 & noLane=13 & noSDBmain=0 & peakhr=PMpk)	THEN	Minor (<30)
8	ELSE-IF	(noLane=12 & SDBmain=minor & road=I495 & incident_type=cpd) or (totalveh=2 & noSDBmain=0 & county=Frederick & incident_type=cpd) or (noLane=12 & noSDBsh=1 & incident_type=cpd & peakhr=PMpk)	THEN	Minor (<30)
9	ELSE-IF	(region=Baltimore & incident_type=cpi & detection=CCTV) or (county=BC & incident_type=cpi & pavement=unspecified & detection =MDTA) or (OC=AOC_Central & totalveh=3 & incident_type=cpi & non-holiday)	THEN	Intermediate (30-120)

BC = Baltimore City; CCTV = closed-circuit television; MDTA = Maryland Transportation Authority; MO = Montgomery; PG = Prince George's.

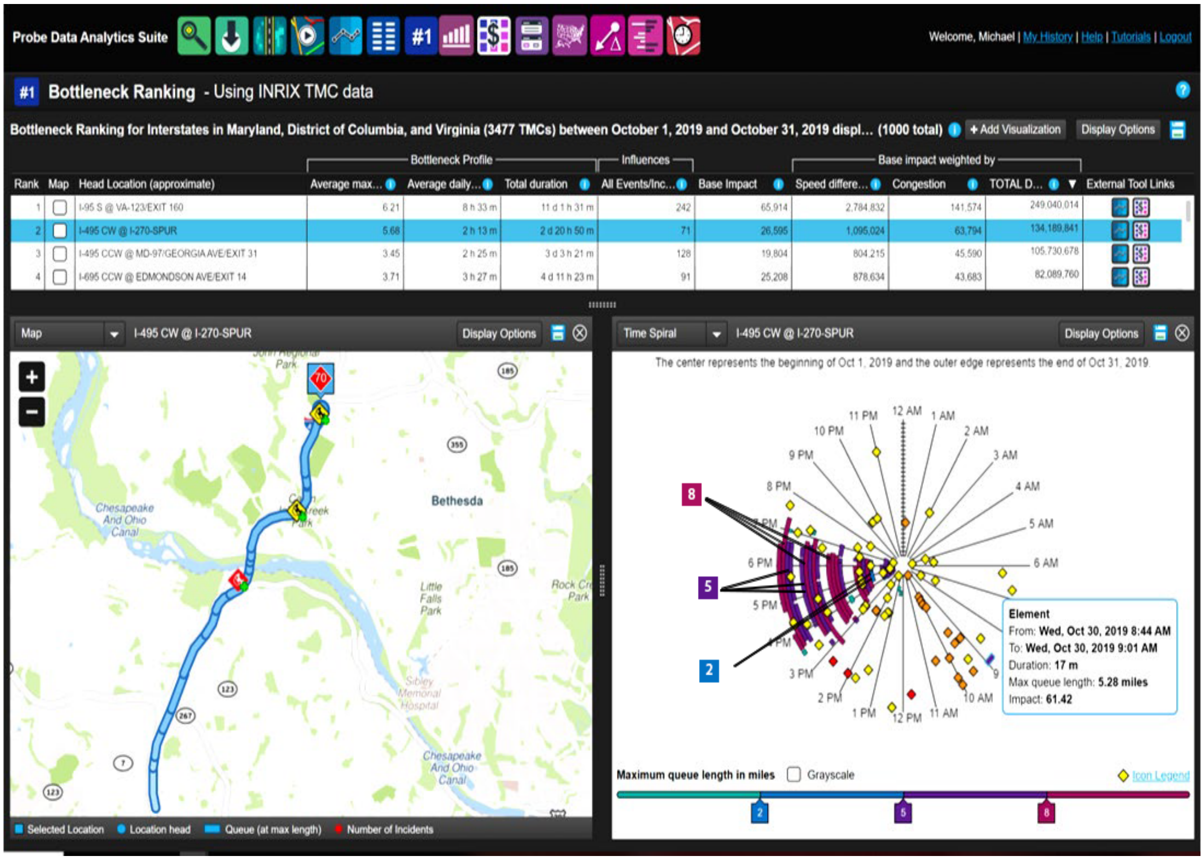
Table 4. Descriptions of variables in an incident duration decision tree.

Variable	Description
Incident_type	Type of incidents: • Disabled: disabled vehicles. • CPI: collision with personal injury. • CPD: collision with property damage. • Fatality: collision with fatality. • Fire: vehicle on fire. • Unknown: no specific information available.
noTT	Number of tractor-trailers involved with an incident.
noPVS	Number of pickup trucks, vans, and sport utility vehicles involved with an incident.
noSUT	Number of single-unit trucks involved with an incident.
Totalveh	Total number of vehicles involved with an incident.
noLane	Number of lanes on both directions (including shoulders and medians).
noSDASH	Number of shoulder lanes on the same direction as where an incident occurred.
noSDBsh	Number of blocked shoulder lanes on the same direction as where an incident occurred.
noODsh	Number of shoulder lanes on the opposite direction from where an incident occurred.
noODBsh	Number of blocked shoulder lanes on the opposite direction from where an incident occurred.
noSDmain	Number of main lanes on the same direction as where an incident occurred.
noSDBmain	Number of blocked main lanes on the same direction as where an incident occurred.
SDBmain	Ratio of number of blocked lanes to total number of lanes on the same direction as where an incident occurred.
noODmain	Number of main lanes on the opposite direction from where an incident occurred.
noODBmain	Number of blocked main lanes on the opposite direction from where an incident occurred.
ODBmain	Ratio of number of blocked lanes to total number of lanes on the opposite direction from where an incident occurred.
OC	Responsible operation center.
Pavement	Pavement conditions: dry, wet, snow/ice, chemical wet, and unspecified.
Chart	1 if CHART is involved in the clearance; otherwise, 0.
Detection	Incident detection sources.
Night	1 if an incident occurs between 8 p.m. and 6 a.m.

Variable	Description
Peakhr	• AMpk: AM peak periods (7:00 a.m. to 9:30 a.m.). • PMpk: PM peak periods (4:00 p.m. to 6:30 p.m.). • Non-pk: Off-peak periods.
Region	• Washington: Fredrick, Montgomery, Prince George's, District of Columbia. • Baltimore: Anne Arundel, Baltimore City, Baltimore, Carroll, Harford, Howard. • Eastern: Caroline, Cecil, Dorchester, Kent, Queen Anne's, Somerset, Talbot, Wicomico, Worcester. • Southern: Calvert, Charles, Saint Mary's. • Western: Allegany, Garrett, Washington.

Performance Measurement Tools

Performance measurement tools (e.g., monitoring and reports) provide regularly scheduled performance indicators for various aspects of the surface transportation system. Performance monitoring tools generate individual measures of performance or summaries of overall performance based on the monitoring of current conditions and historical data. Performance measures or reports (e.g., dashboard summaries) can be used by TMSs as offline DSTs, as illustrated in figure 6. They can help operators identify bottlenecks and areas of recurring incidents, which may require further investigation for mitigation strategies (e.g., signing, parking, and roadway cross section changes).



© 2019 University of Maryland Center for Advanced Transportation Technology Laboratory (CATT Lab). Screenshot created by CATT Lab using the Bottleneck Ranking Tool from the Regional Integrated Transportation Information System Probe Data Analytics Suite.
IMRIX = Intelligent Membership Rating Index

Figure 6. Screenshot. Top 1,000 bottleneck locations in Maryland, District of Columbia, and Virginia between October 1 and October 31, 2019.

Types of Online Decision-Support Tools

Online DSTs are real-time computer-based tools that are well suited to support real-time traffic management and operational decisionmaking. These tools can be separate from or fully integrated with a TMS to provide operators with real-time DSTs to assist with operational decisions. The following sections provide examples of online TMS DSTs that can be fully integrated into real-time TMS operation.

Real-Time Traffic Analysis Tools

As stated in the *Traffic Analysis Toolbox Volume I: Traffic Analysis Tools Primer*, "Traffic analysis tools are designed to assist transportation professionals in evaluating the strategies that best address the transportation needs of their jurisdiction" (Alexiadis et al. 2004).

Traffic analysis tools are designed to help evaluate strategies that address the transportation needs of a jurisdiction. The term “traffic analysis tools” is collective and describes a variety of software-based analytical procedures and methodologies that support various aspects of traffic and transportation analyses. Traffic analysis tools include methodologies such as sketch planning, travel demand modeling, traffic signal optimization, and traffic simulation. Specifically, traffic analysis tools can help operators as follows:

- Improve the decisionmaking process: Traffic analysis tools help decisionmakers arrive at better planning/engineering decisions for complex transportation problems. They are used to estimate the impact of the deployment of traffic management and other strategies and to help set priorities among competing projects. In addition, they can provide a consistent approach for comparing potential improvements or alternatives.

- Improve outcomes of the decisionmaking process: Traffic analysis tools estimate the impacts of various strategies and help decisionmakers set priorities among competing projects. In addition, traffic analysis tools can provide a consistent approach for comparing potential improvements or alternatives.

- Reduce disruptions to traffic: Traffic analysis tools allow decisionmakers to inexpensively estimate effects of traffic management and control strategies, which come in many forms and options, prior to full deployment. Traffic analysis tools may be used to initially test new TMS concepts without the inconvenience of a field experiment.

- Evaluate operational and improvement planning: Traffic analysis tools compare no-build conditions with alternative scenarios that include various types of potential improvements. They report predicted impacts as performance measures, which are defined as differences between the no-build conditions and the alternative scenarios. The reports can be used to select the best no-build conditions or prioritize improvements, increasing the odds of successful deployments.

- Improve design and evaluation time and costs: Generally, traffic analysis tools are relatively less costly than pilot studies, field experiments, or full implementation. Furthermore, they can be used to assess multiple deployment combinations or other complex scenarios in a relatively brief time.

- Present strategies to the public and stakeholders: Some traffic analysis tools create excellent graphics and animations that can demonstrate and/or market what-if scenarios to the public and stakeholders.

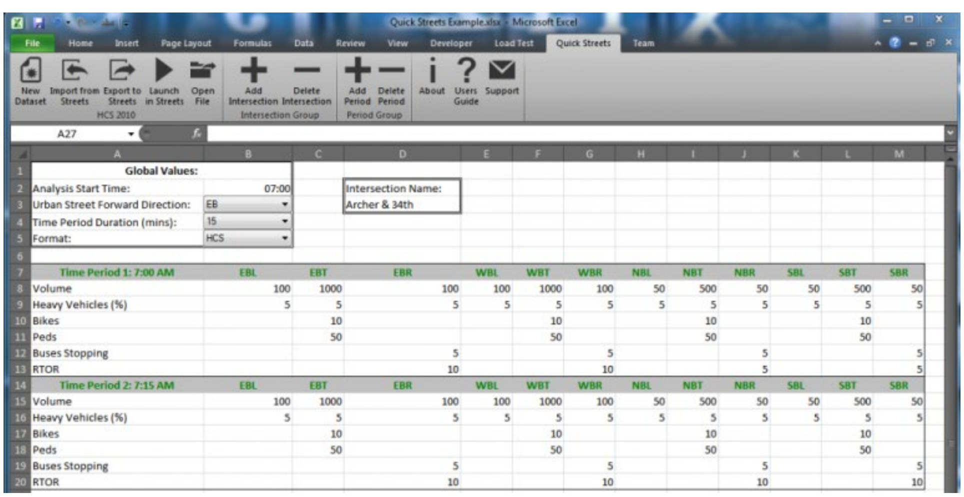
- Manage existing roadway capacity: Some traffic analysis tools provide optimization capabilities, recommending the best design or control strategies to maximize the performance of a transportation facility.

- Monitor performance: Some traffic analysis tools monitor and evaluate the performance of existing transportation facilities. In the future, it is hoped that monitoring systems can be directly linked to analytical tools to create a more direct and real-time process of analysis.

Highway Capacity Software (HCS), the accompanying software for the *Highway Capacity Manual (HCM),* is a common traffic analysis tool used by agencies and transportation professionals (TRB 2016). The *HCM* contains concepts, guidelines, and procedures for computing the capacity and quality of service of various highway facilities and the effects of mass transit, pedestrians, and bicycles on the performance of these systems. HCS implements the procedures defined in the *HCM*, enabling users to analyze various types of transportation infrastructure. Using known traffic volumes and many other inputs, HCS can determine current and projected levels of service (LOSs). It allows agencies to analyze various transportation data and explore possible improvements to their transportation networks. Key characteristics of HCS are as follows:

- It is closed form: Operators input data and parameters, and the HCS process produces a single answer after a single sequence of analytical steps.
- It is macroscopic: Inputs and outputs deal with average performances during 15-min or 1-hr analytical periods.
- It is deterministic: Any given set of inputs always yields the same answer.
- It is static: It predicts average operating conditions over fixed time periods and does not deal with transitions in operations between system states.

Analysis tools in the HCS quickly predict capacity, density, speed, delay, and queuing for a variety of transportation facilities. These predictions are validated with field data, laboratory test beds, and small-scale experiments. An HCS spreadsheet add-in, Quick Streets, converts traffic input data to Quick Streets files, as shown in figure 7. Users can import traffic count data from any source to the spreadsheet to use with Quick Streets. The add-in populates Quick Streets files with multiple intersections, multiple time periods, or both. It allows the user to save input data to a Quick Streets file and launches the data directly in Quick Streets, provided that HCS is installed on the user's computer.



Source: FHWA. Spreadsheet: Quick Streets/HCS and Microsoft Excel.

Figure 7. Screenshot. Quick Streets, an HCS spreadsheet add-in.

Look-Up Tables

A look-up table is a simple tool that helps operators respond consistently to events by providing a table or matrix of conditions and responses. Look-up tables are similar in some ways to incident response plans, which may be offline tools, but look-up tables are routines integrated or coded into a software program and are online tools. As such, they will enable a simplified, real-time process for automating the selection of appropriate plans. Operators enter a few values, and look-up tables use simple algorithms embedded in a software program to calculate the desired output (e.g., plan, action) for a TMS operator to implement. The decision tree example above could be converted to an online look-up table wherein an operator would enter several variables (e.g., location, event type, and number of lanes blocked) into a single screen. The look-up table's software would then calculate an expected duration. The primary difference is that the decision tree would ask one question at a time, whereas the look-up table would ask several questions simultaneously.

Look-up tables are currently used in TMSs for traffic signal timing as part of an incident response plan. When an incident's information is entered into the TMS by an operator, certain basic information is collected through an Event Management screen. A look-up table uses the information provided by the operator, the current time of day, the congestion level or LOS of an intersection, and the direction of a detour route to select a specific timing plan in response to the event. A look-up table can be thought of as a multivariable matrix that is coded into a TMS's software.

Table 5 illustrates the input variables that the TMS's look-up table software would use to select an existing traffic signal timing plan as part of a response plan. The intersection number would be the signalized intersection that is a part of the response plan. The time of day is based on local operations and typical rush hour times. The day of the week could be as simple as weekday and weekend day or specific to each day of the week. Current LOS would be an indication of how much congestion the intersection is currently experiencing, based on traffic engineering principles. Lastly, the detour direction would be the direction that the traffic signal timing plan is trying to optimize for the detour. A look-up table would provide a TMS with an output of timing plan numbers for each combination of variables shown.

Table 5. Input variables within a look-up table.

Intersection Number	Time of Day	Day of Week	Current LOS	Detour Direction
A1	AM peak	Weekday	A, B, C	NB
A2	PM peak	Weekend day	D	SB
A3	Off peak	—	E	EB
A4	—	—	F	WB

— = no data; EB = eastbound; LOS = level of service; NB = northbound; SB = southbound; WB = westbound.

The creation of a look-up table requires the creation of multivariable data within the data subsystem that is used by the TMS software subsystem to “look-up” the signal timing plan number for a specific traffic signal when all other variables are known.

CHAPTER 3. INTEGRATING DECISION-SUPPORT TOOLS INTO TRAFFIC MANAGEMENT SYSTEMS

This chapter provides an overview of the subsystems used to assist in making decisions within TMSs and systems with a TMC. As described in chapter 1, decision support is an overarching concept that embraces all processes and tools that enable better, faster, and more consistent decisionmaking. DSTs include all the various types of resources (e.g., look-up tables and analysis models) that can be used to improve decisions. The use of DSTs online, or into the real-time management and operation of a TMS, requires the integration of these tools into software and data subsystems. The objectives of this chapter are to describe the following:

- Key functions of TMSs.
- Considerations for the integration of DSTs into TMSs.
- Ranges of capabilities for DSTs during integration into real-time TMS operation.

This chapter details key elements of DSTs, their functions, their support for decisionmaking, the data that is used and shared with various TMS subsystems, and the integration of DSTs to support decisionmaking. This chapter seeks to increase its readers' understandings of the capabilities of DSTs, how they may be integrated and used, and their utility in managing and operating a TMS.

KEY FUNCTIONS OF TRAFFIC MANAGEMENT SYSTEMS

Traffic operational strategies encompass a set of functions and actions to improve the management and operation of the roadway network, optimize performance, and improve safety. Deployment and use of operational strategies are informed by agency goals, policies, procedures, and associated performance metrics.

In the early days, freeway-focused operational strategies performed by TMSs primarily focused on performing basic functions such as monitoring traffic conditions and collecting traffic data. As time went on, agencies started using that data to enhance their operational strategies. Functions evolved from basic monitoring and data collection to regulating access, managing incidents, and providing traveler information.

As technology continues to evolve, so, too, does the functionality to collect and manage more complex forms of data. This evolution shapes the way operational strategies are implemented by agencies, as they can leverage more data to make informed decisions. As agencies upgrade their components and software, the timeframe for decisionmaking and responding to changing conditions has improved.

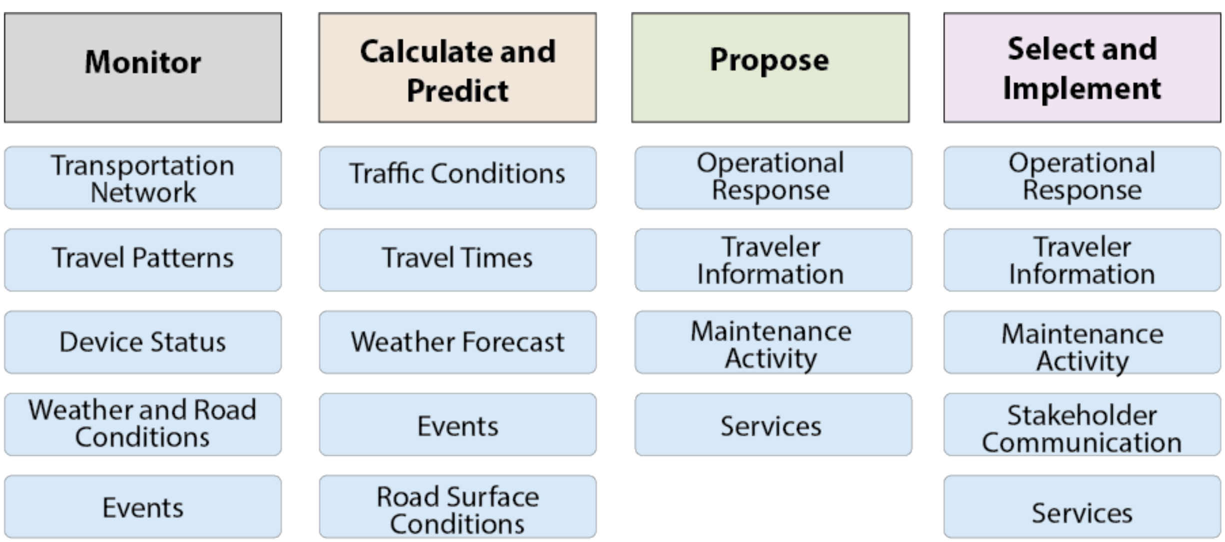
As discussed in chapter 1 on page 2, the decisionmaking framework associated with traffic management involves four decision stages, which have the following functions:

- Monitor.
- Calculate and predict.
- Propose.
- Select and implement.

Operational strategies are enabled by specific functions and actions. Functions that can be implemented to support operational strategies include the following:

- Monitor roadway conditions.
- Collect weather information.
- Perform roadway maintenance during weather incidents.
- Analyze collected data.
- Disseminate traveler information.
- Deploy speed limit reductions or speed advisories.
- Use predictive decision-support software to guide operators in system adjustments and overrides.
- Provide traffic detection and surveillance.
- Manage incidents and special events.
- Manage freeway ramps.
- Manage preferential and priced lanes.
- Coordinate among agencies.
- Monitor and evaluate system performance.

Figure 8 breaks down these functions into subfunctions. The following text then discusses the functions in more detail.



Source: FHWA.

Figure 8. Diagram. Key functions of a traffic management system with decision-support tools.

Monitor Function

The monitor function includes many subfunctions focused on data and information review. During the monitoring process, DSTs monitor data and information available to TMSs to detect changes. When changes occur, DSTs move to the next function, calculate and predict. The following sections outline types of data and information reviewed during the monitor function.

Transportation Network Data

Depending on the field devices used to collect data and the facilities or services being monitored, this subfunction may include monitoring a range of various types of data (e.g., speed, volume, occupancy, travel time, transit vehicle location, and transit service schedule adherence) stored on the data subsystem. New data sources—including connected vehicle data—will provide a tremendous amount of other data (e.g., hard braking, windshield wiper activation). The addition of this connected vehicle data will provide TMSs with additional information on the conditions of the transportation network.

Travel Patterns

This subfunction includes monitoring vehicles throughout a network for origins and destinations, such as transit routes, turning movements at intersections, and transportation flow patterns. This monitoring of routes allows the system to calculate travel times, assess schedule adherence for transit vehicles, and use these data to determine where congestion is occurring.

Device Status

This subfunction includes monitoring the state and status of devices (e.g., DMSs and traffic signals) and assets (e.g., transit buses, pumping stations, and air quality sensors).

Weather and Roadway Conditions

This subfunction includes monitoring travel conditions on roadways, occupancy levels on transit vehicles, and queues at intersections. This subfunction’s process also looks at weather and environmental variables (e.g., moisture or water on roads, wind, precipitation, pavement temperature, and air quality).

Events

This subfunction includes monitoring incidents, construction, and special events within a transportation network (e.g., vehicle crashes, transit bus breakdowns, and construction projects).

Calculate and Predict Function

The calculate and predict function uses a variety of DSTs, which can include algorithms, rules engines, and models, as discussed in more detail in chapter 5. These DSTs are integrated into the software subsystem and use data in the data subsystem to assess current conditions, identify when actions may be needed, and make predictions based on the assessment of data from the monitor function. The DSTs used could be for a specific function or action or could be more advanced and perform multiple different calculations in support of the functions or actions they are designed to support. The following sections outline conditions that can be predicted during the calculate and predict function.

Traffic Conditions

Algorithms and models assess and report on current and predicted future travel conditions (e.g., LOSs, congestion levels, and weather), with and without potential actions, on roadways being monitored. These calculations are usually based on historical information from similar times of day and accepted engineering formulas, and they have the potential to predict the timing and location of weather conditions and crashes (e.g., location, type, and severity) by using models.

Travel Times

Algorithms and software applications (e.g., traffic simulation models) integrated into a software subsystem predict travel times in the future with and without potential actions (i.e., the results of using a specific response compared to no action). Similar to predicting traffic, this prediction function estimates travel times based on current conditions or a future period. These predictions are usually based on historical information from similar times of day, weather conditions, and congestion levels.

Weather

Weather forecasting models (or services) predict current and future conditions for specific locations. For example, a DST in the Dallas ICM implementation used current and predicted weather conditions (e.g., wind, precipitation, and storm direction and speed) on a roadway on a link-by-link basis, which allowed responses to consider the weather for a specific route.

Events

Algorithms and software applications on a software subsystem predict the probability of occurrences of unplanned events (e.g., crashes) and the likely impacts of events. This function typically uses historical crash data to predict probable potential locations for crashes. These predictions are based on current traffic and weather conditions, day of week, severity of any conditions, and time of day. Agencies can use this information to deploy response vehicles preemptively across networks. Additionally, this information can be used to select response plans to unplanned and planned events (e.g., incidents, construction, and special events).

Roadway Surface Conditions

Weather forecasting models (or services), road weather information system sensors, and historical data predict roadway surface conditions. This information is used by roadway maintenance providers to predict weather and roadway conditions and optimize road surface treatments. Treatments can then be timed appropriately, and treatment locations and frequencies can be ranked based on where the effects of weather events are most severe.

Propose and Select and Implement Functions

The propose and select and implement functions involve actions that an agency could take or implement based on information generated during the calculate and predict function—if an acceptance is needed prior to implementation. In terms of the select and implement function, some agencies are able to automate it as their TMSs mature and confidence in their response calculations grow. However, this number remains small. The automation of these functions should be based on the established policies, procedures, expertise, and experience of an agency's operations staff and on the capabilities of a TMS. By applying calibration techniques used in the transportation modeling industry, agencies can develop and calibrate their prediction models and algorithms. These models and algorithms need agencies to routinely monitor, evaluate, test, and make changes as more data are collected and TMSs are operated. The following sections outline primary subfunctions for the propose and select and implement functions.

Operational Responses

Operational responses use predefined and dynamic response plans that suggest actions an agency should use to respond to changes in the transportation network. These response plans usually consist of changing the operation of devices, such as traffic signal timings, DMS messages, ramp metering rates, and lane control indicators. They also include deploying specific assets to the field, such as safety service patrols and temporary DMSs, to address changes identified in the monitor function and initiate responses identified in the propose function. This subfunction can select predefined response plans agencies have previously developed or use rules and algorithms to create a response plan based on its calculations.

Traveler Information

The traveler information subfunction uses various information services to disseminate information to the public regarding transportation network changes and recommended responses. This dissemination can include messages on DMSs, information within agency 511 systems, highway advisory radio (HAR) messages, and various social media messages (e.g., Twitter, Facebook, and other websites). This subfunction also provides special response plans for Amber Alerts, Silver Alerts, and other special events.

Maintenance Activities

The maintenance activity subfunction uses outputs from the weather and road surface condition subfunctions (in the calculate and predict function) to implement specific maintenance activities related to weather, such as snow removal, salt applications, and road closures. It also implements actions for other environmental incidents, such as alerts for poor air quality or high water levels. For some systems, this subfunction could also identify other required maintenance activities, such as preventative or emergency maintenance for devices, maintenance for roads, and maintenance for transit vehicles.

Services

The services subfunction uses output from the calculate and predict functions to request services from entities outside of a TMS. The services requested could include wrecker services, emergency services, and other services needed to fully respond to an event managed by a TMS.

Stakeholder Communication

The stakeholder communication subfunction uses various communication mechanisms to inform stakeholders about TMS actions. These mechanisms include stakeholder requests for TMSs to perform actions as part of ICM programs.

INTEGRATING DECISION-SUPPORT TOOLS INTO TRAFFIC MANAGEMENT SYSTEMS

One of the objectives of integrating a DST into a TMS is to automate or support the decisionmaking required by operators or a TMS to actively manage and operate a transportation network. Any DST integrated into a TMS requires an interface and procedures for an operator or analyst to integrate the DST. The DST (and/or its software) may need to be integrated with a software subsystem, data subsystem, computing hardware, and DST-generated data users or decisions (e.g., data must be translated into appropriate formats, if needed). This integration process enables sharing and using the data generated by the tool. The following sections discuss issues to consider with these key interfaces.

The Data Subsystem

The core functions of the data subsystem are to provide data processing and storage for a TMS and support access to data by other subsystems and external users. A data subsystem uses application programming interfaces (APIs) to interface with other subsystems or components that extract, send, and enable the data to be transformed and loaded into the database in the TMS. An API specifies how software programs should interact, what data they should exchange, and how the data are exchanged (i.e., format and type).

The data subsystem also receives data from external sources by using APIs or interfaces to receive the data, translate it into appropriate formats, and save it. Once the data are received, the data subsystem uses tools to monitor and manage the subsystem so that data use is efficient, timely, and easy to maintain. The data subsystem needs to have the following functionalities to support any DST integrated into the operation of a TMS:

- Retrieve data from all sources available to the TMS in a timely fashion, as they become available—including real-time data streams and less frequently or even manually updated data.

- Catalog data received in an electronically accessible data catalog.

- Transform data received into a format suitable for storage, further processing, and retrieval by users and other applications.

- Secure data received so that they are only accessible according to the usage policy and authorization parameters of an agency.

- Manage user and application accounts, authentication, and authorization for accessing the data subsystem.

- Encrypt communications and data between the data subsystem and its users.

- Provide access to all data as appropriate to the user making the request.

- Provide data to authenticated and authorized users in a timely fashion, including data in storage and real-time data streams.

- Sustain availability and performance level necessary to support TMS operations.

- Produce and provide status and diagnostic information to support the operation, maintenance, and management of the data subsystem.

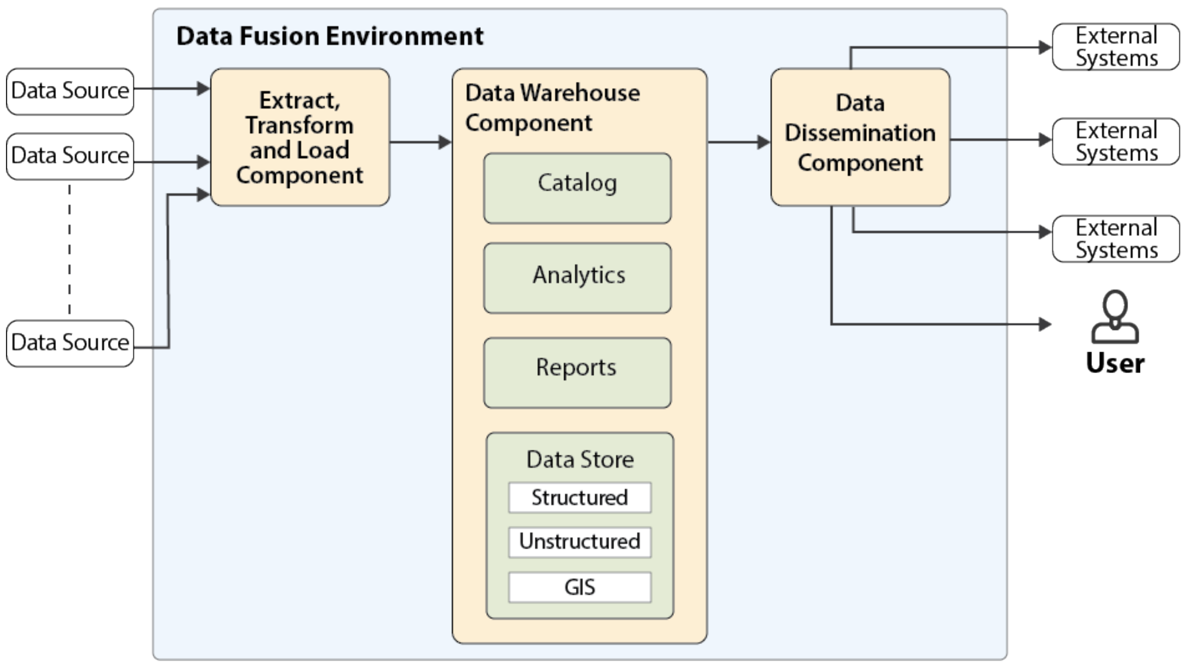
Interfaces With Software for Decision-Support Tools

The TMS software subsystem uses software APIs to integrate software programs installed on this subsystem or to share data with other subsystems. An API is a description of the routines, protocols, and tools for interfacing and exchanging data with a software application or program. An API specifies how software programs should interact, the data they exchange, and how the data are exchanged (i.e., format and type). There are two types of APIs or interfaces typically developed and integrated into the software subsystem, as follows:

- Data providers (provide data to the TMS): The provider usually dictates their interfaces and the processes, protocols, and requirements (e.g., formats) for receiving and using data from their system. The provider must furnish the associated API documentation (e.g., data dictionary, release notes, configuration guide, and user guide) to follow.

- Data subscribers (receive data from the TMS): The TMS dictates these data interfaces, and the subscribers must develop their interfaces to meet the requirements specified for the appropriate processes, protocols, and formats. The TMS must provide the subscriber with an associated schema or data definition.

Figure 9 illustrates an example of a data subsystem that includes multiple databases with various sources generating and submitting data, which are extracted, transformed, and loaded into databases within the data subsystem. An API is developed for each data source, as each source will have different data in different formats and will potentially use different protocols to provide their data. Each API communicates with the data source, extracts the data, translates it from the data source to the TMS system format, and then stores it in the TMS database. For the TMS to provide data to other systems, an API is developed and used to interface with each TMS subsystem or other systems external to the TMS, where these systems could be subscribers and providers of data. For instance, the data subsystem may have a database specifically for storing traveler information that a statewide 511 system would like to use. The 511 system would use the API to connect to the data subsystem and extract that information. Within the 511 system, an API would also be needed and would be integrated with its software to translate the data from the TMS database format to the format of the database that the 511 system uses.

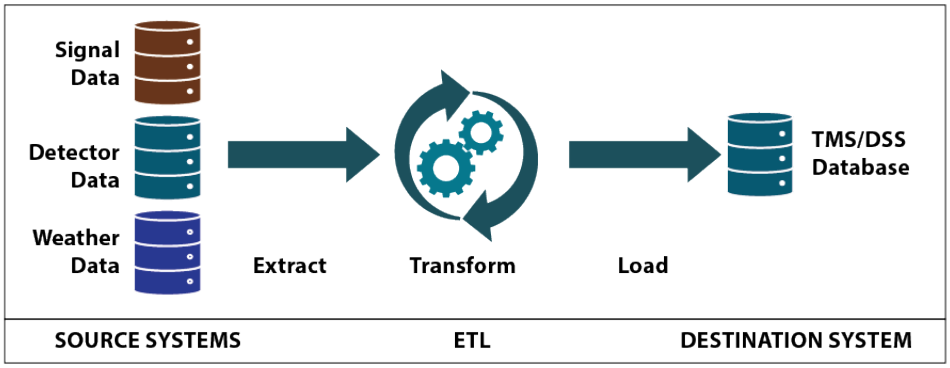


GIS = geographic information system.

Figure 9. Diagram. Sample data subsystem.

Figure 9 includes neither the use of the data nor the potential to control the field devices; rather, it is an example of the process within a TMS for carrying out functions supported by the software for a DST integrated into the software subsystem. The analytics and reporting functions shown are typically separate software that interface to the database to use and analyze the data for various reports and dashboards.

APIs are developed and installed between the DST and TMS subsystems or field devices to exchange data, issue commands, and integrate the operation of each system. Figure 10 illustrates the process to extract, transform, and load (ETL) data from external sources into a specific database on the data subsystem. The source systems provide APIs to developers, who develop the tools, processes, and other information that use or incorporate the data.

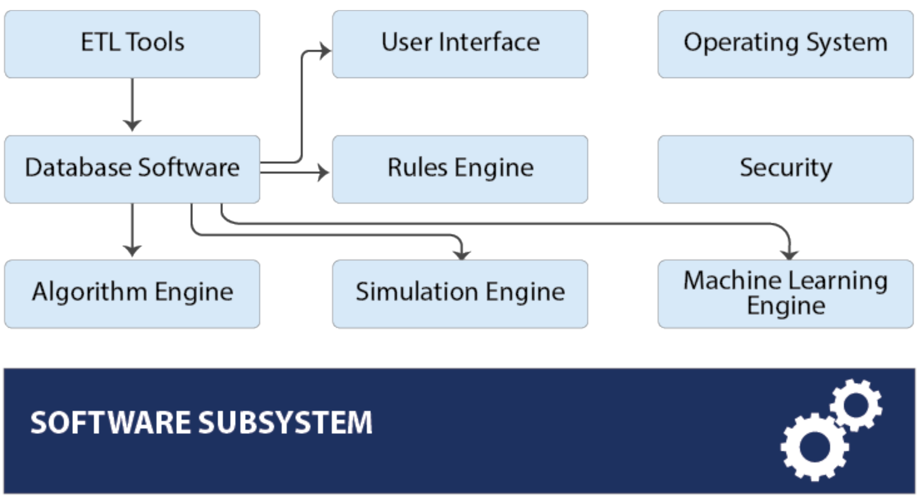


Source: FHWA.

Figure 10. Diagram. Extract, transform, and load process.

The Software Subsystem

The software subsystem includes the programs that support the functions and services of the TMS. This subsystem will share some software products with the entire TMS, and specific software programs installed for other subsystems or DSTs only (e.g., the various engines shown in figure 11).



Source: FHWA.

Figure 11. Diagram. Software subsystem and installed software.

Assorted types of software programs and APIs are integrated into this overall software subsystem to carry out all the management and operating requirements of the system. A software subsystem could include multiple different software programs (e.g., COTS software and proprietary), as shown in table 6.

Table 6. Software for various types of decision-support tools.

Software Products	Knowledge-Driven	Data-Driven	Model-Driven
Rules engine	Yes	Yes	Yes
Algorithm	Yes	Yes	Yes
Simulation software	No	Yes	Yes
Machine learning	Yes	Yes	Yes

These software products can be divided into two main categories: Software products that are typical to all TMSs and software products that are unique to DSTs. The following sections detail these two categories.

Software Products Typically Used in TMSs

The primary types of software that are typically used in TMSs include software needed to host a TMS on computing hardware and some software with assorted specialized functions, as follows.

ETL tools: Provide functions for extracting data from data interfaces, transforming them into the format needed or required for the data subsystem, and then loading them into a database to save. Many ETL tools were originally developed to make the tasks of saving, accessing, and using data easier. ETL tools spare developers the arduous task of handwriting structured Query Language Code and replacing it with easy drag-and-drop functionality to develop or make changes to a database when new data sources are introduced.

Operating system software: Manages computer hardware, subsystems, and software programs installed on the software subsystem and provides common services to support managing, maintaining, and operating the software subsystem.

Database software: The database management system, which is a tool that makes it possible to organize data within a database. Several COTS and open-source products are used in TMSs today.

Security: Software that manages user profiles and access to a TMS. This software is sometimes a part of the operating system or of an agency's larger network (i.e., active directory). Other times, a TMS will have its own security software and user permissions. A DST may require separate, dedicated security software to manage users and access to its other software components.

User interface (UI): A software tool that provides users with a simplified, user-friendly way to interact with a TMS and DST. It will display information on transportation network status, events, recommended responses, and expected performances with and without recommended responses.

Software Products Unique to DSTs

The four main types of software used when integrating a DST or its associated software program are described in this section. Based on a literature review, table 6 illustrates which of these types of software may be used in the three classifications of DSTs described in chapter 2. These software systems should have the ability to allow agencies to easily add onto them or to make changes as needed. The ability to easily make these types of modifications helps to ensure that agencies can continue to modify and evolve the capabilities of their systems as demands and needs change.

Rules engine: Software program or code integrated into or integrated via an API to execute one or more business rules that are used to make decisions in a DST. The rules might originate from external laws and regulations, agency policies, or other sources. The rules are configured and integrated into the rules engine software program to execute required functions or prepare information needed to support decisions. A rules engine should be used when the following four conditions occur:

- An agency has developed complex scenarios with simple rules.
- An agency has not identified an algorithmic solution.
- A TMS has endured ever-changing scenarios.
- An agency has needed to make decisions quickly, usually based on partial data.

A rules engine should not be used when the following three conditions occur:

- The project involves very few self-contained rules.
- The business logic changes rarely.
- The application demands rigid control of the execution flow.

Several commercial products on the market can be used by systems developers or agencies to develop a rules-based DST without having to completely develop custom code. These tools will still require some configuration and integration into agency systems to provide the functionality desired by operating agencies.

Algorithm: Algorithm software utilizes logical and mathematical formulas used in a DST to calculate various outputs needed by the DST. The algorithms are usually based on specific formulas and may include engineering (e.g., traffic flow and environmental calculations), statistical (e.g., averages and means), and financial (e.g., cost and benefit) calculations. For example, as discussed in chapter 2, HCS uses industry-recognized formulas and logic to process data and calculate results used for operations. Some agencies have integrated HCS into their TMSs for various traffic-engineering analyses in real time and nonreal time (e.g., signal timing analysis). Using an API, the HCS software receives data from a TMS requesting the HCS calculate a specific formula. Once the HCS calculates the result using an API, it provides the TMS with the result requested.

Simulation software: This software uses calibrated models to determine how the transportation network will perform based on various potential actions. Simulation can be used in real time to determine effects on travel time, various environmental impacts, and person and vehicle throughput. Simulation tools integrated into a TMS as part of a DST can enhance an agency's ability to analyze strategies and perform complex data calculation in real time. In certain cases, simulations can also offer predictive capabilities. Online modeling tools can be an effective way to perform such functions and, in turn, are desired, although they are not strictly required. These tools can be expensive to purchase, set up, and maintain; however, if used correctly, they can provide many benefits. As confidence in the simulation tool's results increases and agencies become more comfortable with its recommendations, a higher level of automation among TMS functions can be considered.

Machine learning: Machine learning is based on the idea that data can be analyzed to identify patterns, and software can learn to make decisions based on the patterns within that data with minimal human intervention. Machine learning software uses various formulas that software developers and data scientists establish and integrate into it. The software uses these formulas to recognize patterns and learn how to optimize their use based on historical data and patterns. As more data increasingly become available, the quality and accuracy of the recommended or automated actions using the formulas also improve. These tools can be challenging to set up and maintain due to the amount of data required to calibrate the learning algorithms; however, if calibrated correctly and monitored, they can provide many benefits and lead to a higher level of automation among TMS functions.

Computing Hardware

A computing hardware subsystem can vary greatly depending on its purpose and location. A DST or its accompanying software program may be a part of a field device, traffic controller, TMS, or its TMC. The computing hardware subsystem for a TMS is usually shared to provide economies of scale and reduce overall cost. Most agencies use typical IT servers to host the software and database subsystems discussed previously. The computing hardware includes servers with necessary processors and memory, network communication equipment (e.g., routers and switches) to enable data exchange, and data storage (e.g., storage area networks and hard drives) to store the software and data within the subsystem.

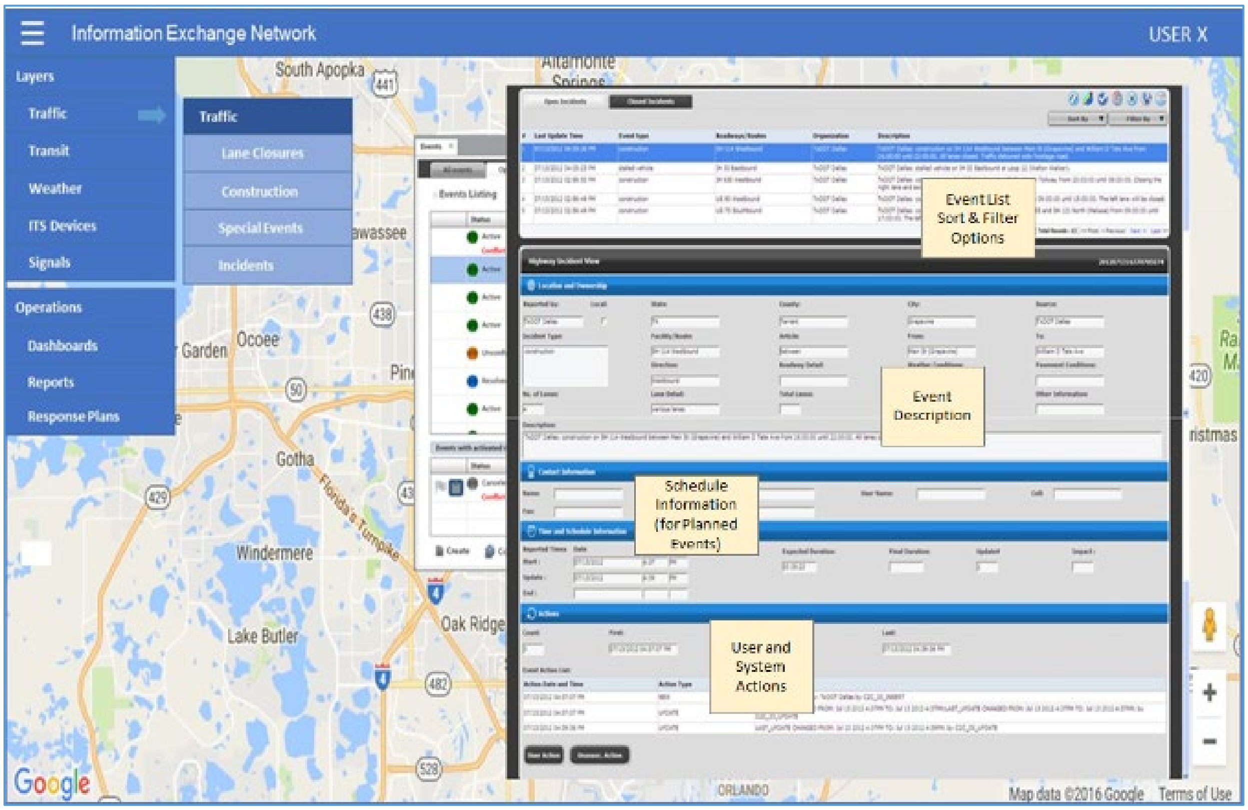
The capacity and performance requirements of the computing hardware are directly related to the functionality of the TMS, the processing power needed, and the amount of data involved. When considering a computing hardware, agencies need to determine the requirements of the TMS and of any specific DST, including processing and memory needs. This report does not offer guidance on how to select the appropriate computing hardware components; rather, agencies should use their IT department standards and sizing methodologies to select from existing options.

User Interfaces Specific to Decision-Support Tools

A UI provides a visual means of interacting with a TMS, using items such as windows, icons, and menus that are commonly used by most modern operating systems. Depending on the function of a DST, the management of its software or algorithms, and the utilization of tools by the TMS, various UIs may need to be developed. The UIs may need to be developed separately, or the TMS may need to be modified to integrate these UIs into it.

For example, the Dallas ICM project consisted of three UIs, because three separate pieces of software were developed by three different companies. The primary interface was the operator UI, which was a modification to the existing TMS. This operator UI allowed operators from all agencies to view transportation data in the region and manage events cooperatively through a recommended response plan. The TMS also had two administrative interfaces—one to manage rules and algorithms for selecting response plans and another to manage the predictive model. Ideally, the TMS would be modified to provide a single UI.

The purpose of the graphical UI (GUI) shown in figure 12 was to provide a web-based information exchange tool for stakeholder agencies to share information and manage incidents, construction, and special event information. In simple terms, the GUI is the presentation layer for the TMS. The Dallas ICM GUI provided agency users with a graphical tool to manage and monitor the status of their transportation networks, giving them full event-management capabilities and allowing them to make informed decisions regarding the management of their transportation infrastructure.



Screenshot © 2015 Dallas Area Rapid Transit. Background map © 2016 Google® application.

Figure 12. Screenshot. Dallas integrated corridor management project operator user interface.

The main functionalities of a TMS's GUI software program are as follows:

- To provide stakeholders with the ability to exchange data regarding incidents, construction, and special events in an interactive manner.
- To include an event management module to allow stakeholders and partner agencies to create incident or planned event trackers within the system and manage these events from detection to resolution.

- To provide the current status of devices and roadway and transit networks within the corridor on a map and through lists.

- To provide incident response plan information to corridor stakeholders.

- To monitor the status of recommended and implemented response plans from plan recommendation to incident resolution.

Figure 13 illustrates two rules-engine administration interfaces for the Dallas ICM TMS. These interfaces were used by an administrator to view current events and data received by the system on a data input screen (top left screenshot) and determine whether they match the rules within the rules engine on a rules evaluation screen (lower right screenshot).

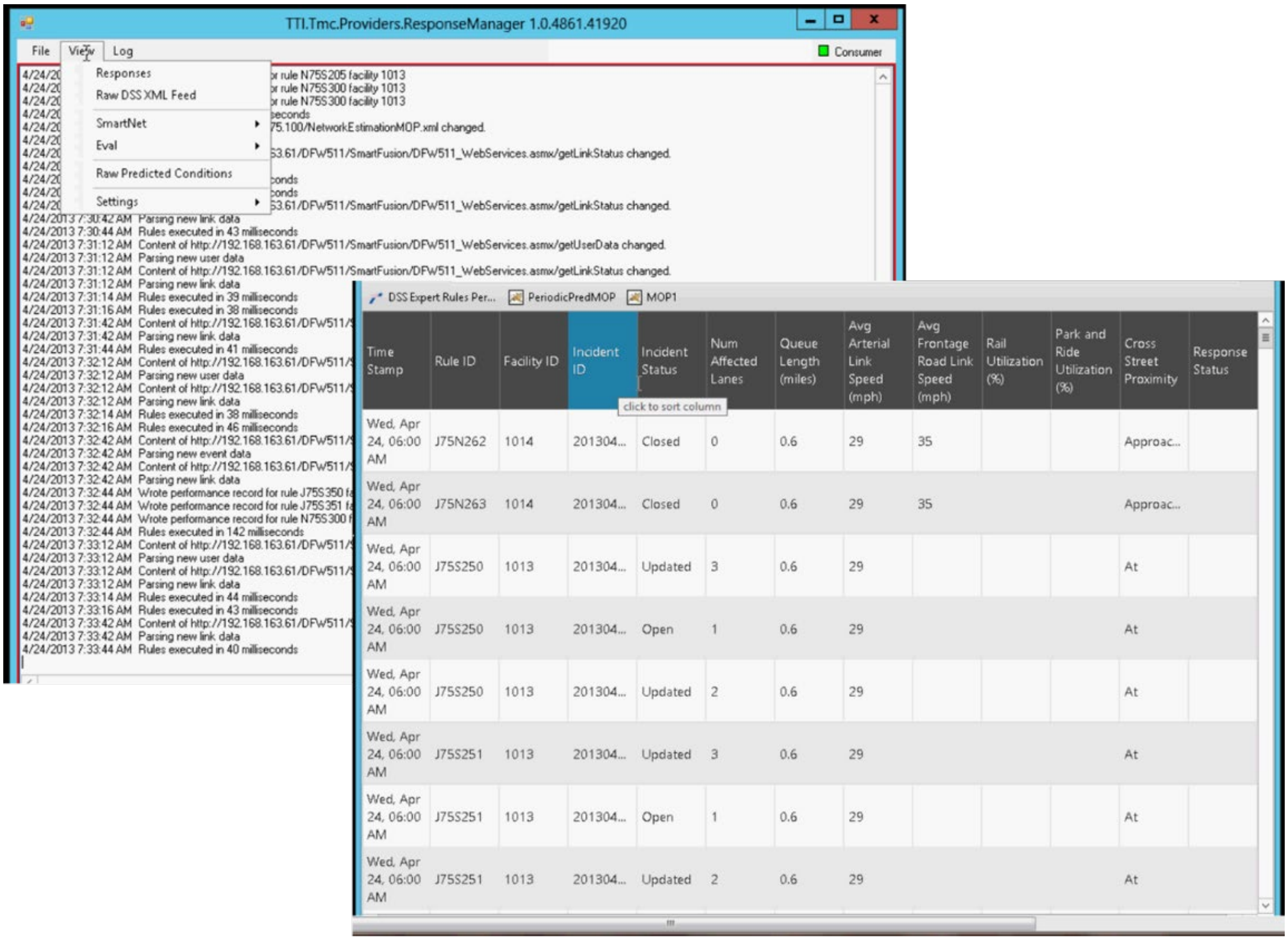


Figure 13. Screenshots. Dallas integrated corridor management project business rules component user interface.

The data input screen provides the following information:

- A time-stamped list of the event data received by the rules engine.
- The evaluation performed by the engine.
- The actions recommended by the rules engine.

The rules evaluation screen details active events within the TMS and states which rules within the rules engine should be applied. Specifically, this screen lists the following:

- Location of event (Rule ID translated from the actual roadway link).
- Number of lanes affected (percentage of lanes affected based on total number of lanes).
- Length of queue (miles of backup from incident location).
- Average speed on roadway and potential diversion routes.
- Transit and park and ride utilization (percentage full).
- Response status (if the event and rules were met and a recommendation was made).

A DST administrator used the information provided in these interfaces to troubleshoot errors and analyze why the DST selected a response to ensure the DST was operating correctly. This information was used by the agencies at their monthly operations meetings to decide if changes to the rules were needed and to provide metrics on the use of the DST (e.g., number of events where it was used).

The Dallas ICM TMS also used simulation software to predict the performance of recommended response plans from the rules engine versus do-nothing responses. Figure 14 illustrates the interface for the simulation software’s assessment of two-such scenarios. Once the simulation software modeled a recommended response and a do-nothing response, it provided performance measures for both. If a recommended response plan improved the projected average travel time of a vehicle through the corridor by more than 2 percent, the response plan was recommended; otherwise, a do-nothing response was recommended.

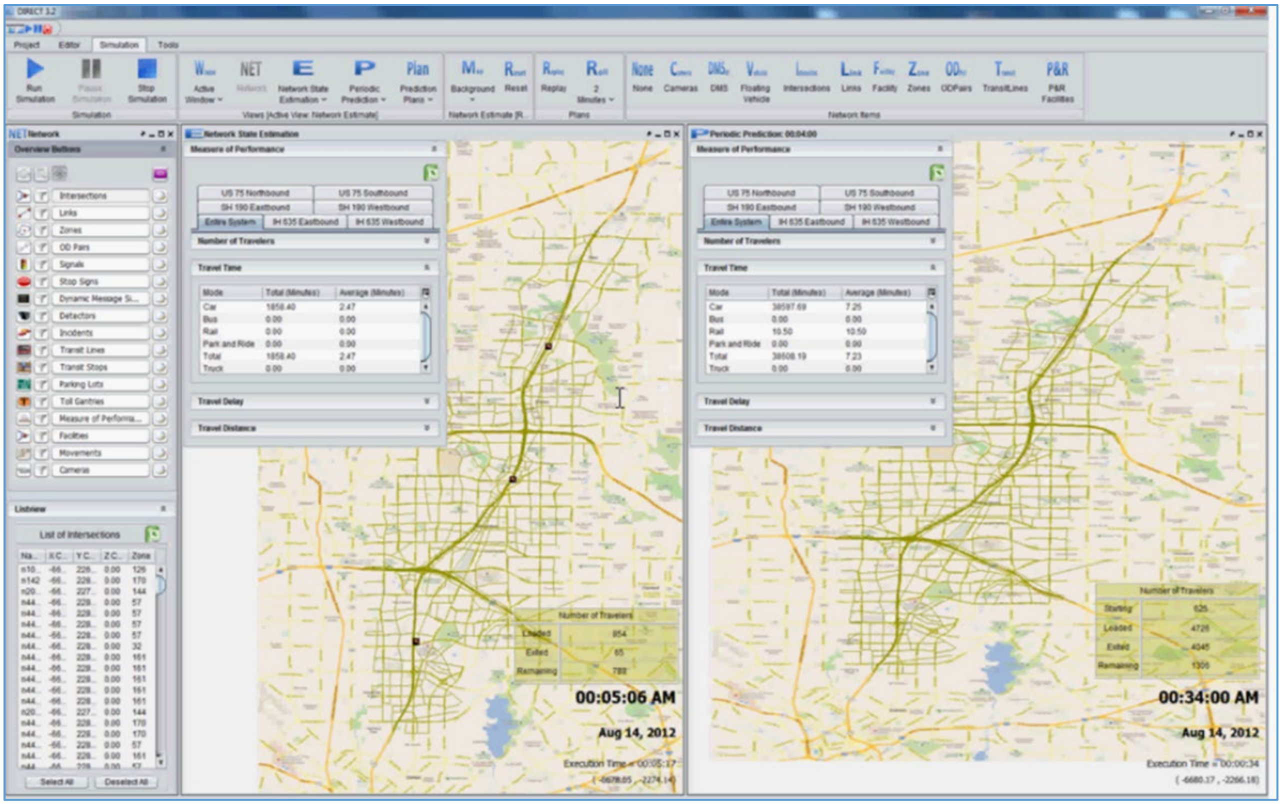


Figure 14. Screenshots. Dallas integrated corridor management project prediction component user interface.

Logical Architecture of a Traffic Management System

Fully integrating a DST into a TMS provides a single software program to support all the functions of the tool. The advantage of integrating a DST into the software subsystem of a TMS is that a smaller computing hardware footprint can be used, maintained, and managed. Depending on how a TMS is structured or designed, upgrades and changes to the software program or API for a DST are only needed in one location. If a DST is not located in the software subsystem, any upgrade to this subsystem will require revisions to the DST's software and all of the APIs; this will ensure the data will continue to be exchanged and used through these interfaces.

One example of a DST that is fully integrated into a TMS system is the Maryland CHART TMS. The goal of the CHART system is to manage freeway and arterial traffic flows more efficiently and safely. When freeways and other primary routes are unexpectedly congested, the TMS recommends response plans based on business rules. These response plans include changes in DMS messages, detour routes, and traffic signal timing plans. Arterial signal systems provide remote and adaptive traffic signal control and coordinated signal timing. Traffic signal technicians and CHART system operators can better balance demand and capacity by adjusting traffic signal timing remotely through the TMS. The advantage of this system is that different operators will respond to incidents in a consistent manner, regardless of their experience levels—a new operator will be able to respond with the same response plan as a veteran operator with years of experience. By integrating the DST with the CHART TMS, the Maryland State Highway Administration can reduce the computing hardware needed, only employ a single systems developer, and more easily make system updates to the TMS.

An example of a fully integrated software subsystem is the Australian VicRoads TMS called STREAMS®, as shown in figure 15. STREAMS implements a feature called "Dynamic Plan Selection." Depending on the density of traffic and the dominant direction of traffic (e.g., inbound, outbound, or bidirectional) on a road, nearby signalized intersections are operated using several predefined traffic plans. A user sets up the signal timing and picks the traffic density levels and directions that apply. The system then automatically selects the appropriate signal timing. When a time-based traffic plan schedule would be inappropriate because of varying traffic levels, it automatically adapts the selection due to unexpected traffic levels. The STREAMS TMS uses many of the same functions as CHART, the fully integrated DST discussed previously, but it uses a real-time model to calculate congestion, travel times, and other conditions for its surface transportation network.

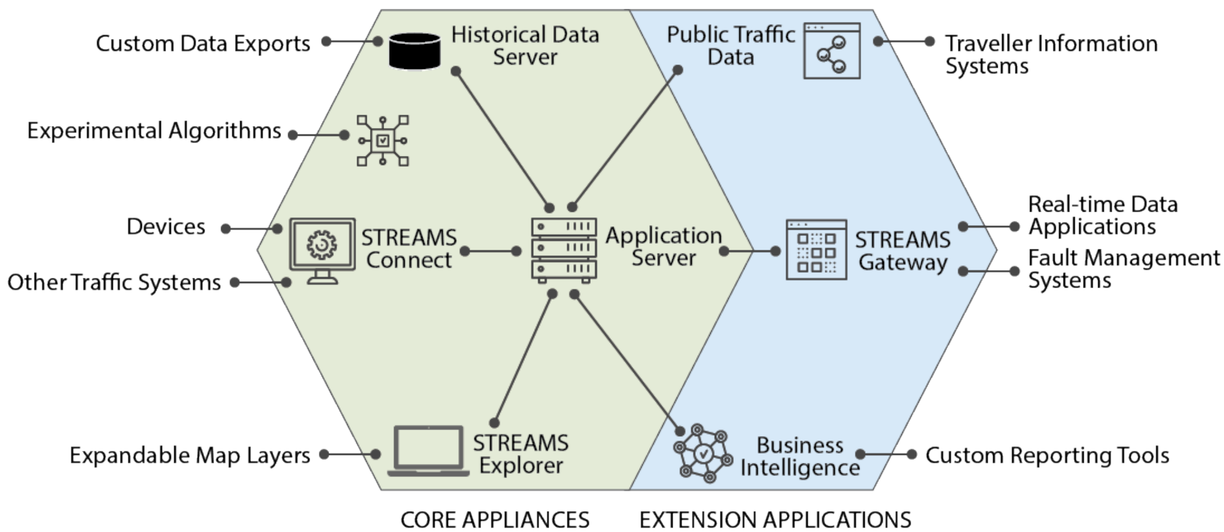


Figure 15. Diagram. VicRoads traffic management system software subsystems architecture.

STREAMS is fully integrated, providing a shared data subsystem, computing hardware, and UI. The software is integrated into the TMS so that a single, fully integrated TMS is used by VicRoads. As with the CHART TMS, a key advantage of the integrated STREAMS TMS is the reduced cost of the computing hardware, data subsystem, and ongoing operations and maintenance. The STREAMS software provides each of the four functions discussed earlier in this chapter, as follows:

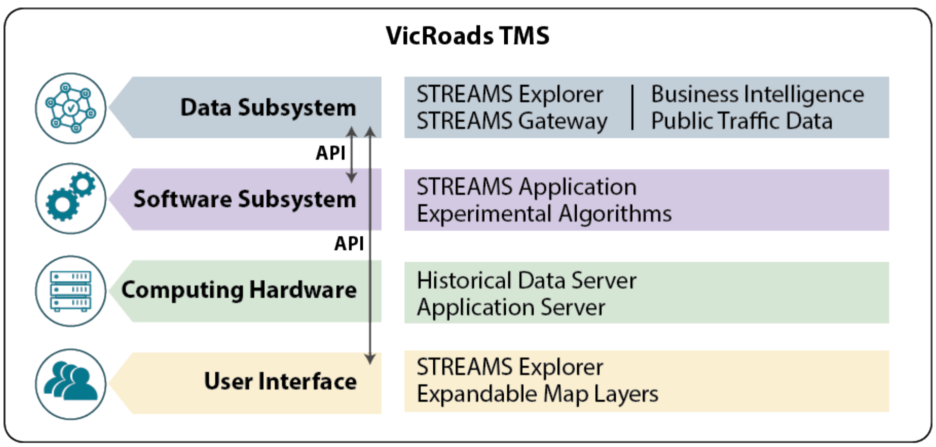
- Monitor: STREAMS collects and processes data from various systems, including devices and other traffic systems (shown on the left side of figure 15). It analyzes this information to determine when the data have changed and compares it to the data in the historical data server.

- Calculate and predict: STREAMS uses traffic signal algorithms and models to make traffic signal timing calculations and recommendations.

- Propose: STREAMS provides recommendations to the TMS operator to improve the performance of the transportation network based on outputs of traffic signal timing plans that were selected or created within the calculate and predict function.

- Select and implement: STREAMS changes signal timing plans in affected traffic signal controllers, deploys messages on DMSs, and provides traveler information through various media automatically when a TMS operator approves proposed actions.

The architecture shown in figure 15 is mapped directly to the subsystems previously discussed, as shown in figure 16.



Source: FHWA.

Figure 16. Diagram. Mapping the VicRoads traffic management system to the four key subsystems of traffic management centers.

The STREAMS data subsystem includes APIs to receive data from external systems and field devices (via STREAMS® Connect) and provides data to those systems through the STREAMS Gateway®, business intelligence, and public traffic data components shown in figure 15. The software subsystem consists of the STREAMS application and experimental algorithms. The STREAMS application is composed of the TMS’s capabilities; meanwhile, the experimental algorithms provide additional decision-support software functions. The computing hardware consists of the application server and the data server. The data server provides some of the computing hardware for the data, and the application server provides the computing hardware for the TMS. The UI provides a map-based interface for the operators to monitor, control, and manage traffic on the streets and highways. The STREAMS subsystem UI is called STREAMS® Explorer.

CHAPTER 4. TRAFFIC MANAGEMENT SYSTEM DECISION-SUPPORT NEEDS

While previous chapters defined the key elements and functions of TMSs, this chapter begins the process of planning for and integrating DSTs into a TMS. It examines needs, requirements, benefits, and related issues to consider when integrating a DST into a TMS and its TMC. The objectives of this chapter are as follows:

- To describe issues agencies should consider before moving forward with making an investment in a DST.
- To frame needs for decisionmaking in the context of overall TMS system planning.
- To describe needs assessment strategies that can be used in assessing specific decision-support needs of a TMS.
- To present a process for evaluating needs in order to develop DST requirements.

This chapter introduces the issues an agency should consider as it evaluates its TMS capabilities, plans to improve its system, and considers its decisionmaking needs. It also details institutional issues to consider in these processes and in making decisions for TMSs.

After reading this chapter, the reader should better understand the needs and implications involved in integrating DSTs into TMSs. This chapter seeks to support agencies in identifying issues to consider when integrating DSTs into TMSs.

CONSIDERATIONS BEFORE PURSUING A DECISION-SUPPORT TOOL

The FHWA *Systems Engineering Guidebook for Intelligent Transportation Systems* says the following:

> "Needs assessment is an activity accomplished early in system development to ensure that the system meets the most important needs of the project's stakeholders. The goal is to ensure that their needs are well understood before starting development. In many cases, there will be more needs than can be met, even conflicting needs." (FHWA 2019)

When deciding if an agency should implement a DST, its operators should judge whether there are issues they are trying to solve that the current TMS does not address. The questions framing an agency's considerations prior to pursuing a DST should include the following:

- What needs or issues is the agency trying to solve?
- What constraints is the agency facing in implementing a solution?
- What process is the agency using to make decisions, and what information and performance data are being taken into account as a basis around which other alternatives can be considered?

The agency should assess situations where decision support would possibly be beneficial and/or steps in the current process where there are bottlenecks or potentials for human error. It should also identify and assess how decisions are currently being made and evaluated and identify any information and tools being used to support decisionmaking. This process can lead to identifying those elements in an agency's decisionmaking process where the appropriate support tools, including a DST, could potentially assist users in their task flow.

UNDERSTANDING NEEDS IN THE CONTEXT OF OVERALL SYSTEM PLANNING

The processes for determining needs; assessing feasibility; and then planning, designing, and implementing a DST may seem complicated or confusing. However, by following a systems engineering process and answering questions thoughtfully at each step, one can determine the best approach for designing and implementing a DST with relative ease. Systems engineering is an organized approach to developing and implementing a system that can be applied when developing any type of system. The International Council on Systems Engineering (INCOSE) defines systems engineering as follows:

> "[Systems engineering is] an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem.
>
> "It is crucial to use the systems engineering approach in designing ITS infrastructure so that the technology effectively supports the management and operation of the transportation system. A systems engineering analysis is required for all ITS projects using Federal funds, per Title 23 Code of Federal Regulations 940.11." (INCOSE 2012)

Assessing Traffic System Needs

Assessing needs is an essential part of the overall planning, design, and implementation process for a DST. The key steps are as follows:

Assess current capabilities and performance (get the "lay of the land"): Usually, this step begins with gathering information about the overall landscape of the traffic system being managed, including current capabilities, available data sources, and desires for enhanced capabilities. This step involves gathering input from various stakeholders and examining their complaints and concerns about certain decision choke points that indicate problems.

Identify and evaluate user needs and analyze gaps: Building on the preliminary information gained in the first step, conduct a more detailed stakeholder needs assessment and gap analysis to identify shortcomings. Needs assessment should set aside preconceived notions of what a DST will do and elicit stakeholder needs, desires, and constraints.

Develop a concept of operations (ConOps): Based on user needs, a ConOps frames the overall system, including specifying issues to be addressed by the DST. It identifies the stakeholders, elements, and capabilities of the planned DST. It also establishes relationships among the stakeholders and system elements, including the flow of information and processes to be performed.

Translate user needs into system requirements: This stage is very familiar to transportation engineers and entails mapping out potential specifications and operating procedures and formalizing expectations regarding the performance of a DST.

Design the DST: Define a design to implement DST requirements. The design is typically laid out in two successive stages: high level and detailed.

Develop or procure the DST: Implement a DST in accordance with the detailed design specifications. This process involves hardware and/or software development, system testing and verification, integration, and so on.

Begin iterative monitoring, evaluating or testing, and updating of the DST: The implementation must be tested and updated. The need for updating external systems that will interface with the DST is often overlooked but is important. A formal maintenance process should be implemented to manage updates needed for the DST software and any APIs that may be used.

A development team should evaluate the following key high-level considerations in determining the appropriateness of a DST:

- A DST frequently suits environments requiring structured, rapid-fire decisionmaking, particularly multitiered decisionmaking across a range of transportation modes and agencies.

- A DST offers a mechanism for managing, harnessing, and optimizing the power of information as ever-growing volumes and varieties of data (of increasing complexity) become available to transportation operators.

- A DST helps in facilitating fast, structured, objective decisionmaking that builds on the experience of experts and promotes collaboration across multiple agencies, transportation modes, and stakeholder groups.

- A DST potentially helps in overcoming the risk of unintended operator bias in actions and decisionmaking because of its reliance on empirical data and structured rulemaking.

- A DST potentially assists in predicting adverse transportation conditions and identifying corresponding mitigation strategies if it is suitably designed and implemented.

After determining that a DST makes sense, the development team should commence a more in-depth evaluation of circumstances and conditions by examining the following issues:

- What level of implementation would the DST function at?

    - Levels of implementation can correspond to offline strategic uses for long-term decisions, which, historically, were the traditional use of DSSs that may not have included TMS data (e.g., asset management and traffic analysis).

    - Online, real-time traffic management decisions can be used as integrated tools within a TMS, whether perpetuated by a TMS, device, or TMS operator.

- What roles and responsibilities would system users undertake, including agency personnel (operators, managers, etc.) and those at partner and stakeholder organizations?

- What modality would present information to operators and other users? Would information be presented using visual displays, through alerts, or by other means?

- What timeline would decisionmaking follow? What time demands would system users have? How much time is available to implement recommended actions? Are there gaps between the time it currently takes to make decisions and the timeframe within which those decisions are actually needed?

- What types of data, information, and requirements would the DST need for the achievement of desired function and performance? How detailed would recommended actions communicated to users need to be?

- What would the expected benefit of the selected DST be, given data required and improvements in performance, versus other types of DSTs that could be used?

- What would the return on investment be for the expense of the development, implementation, management, and usage of these data? Is the DST worth the added expense?

NEEDS ASSESSMENT STRATEGIES

A needs assessment is a systematic process for understanding and evaluating underlying gaps between desired and actual conditions. It can be used to plan for a new system or process, as well as to improve current processes and activities. This section discusses some generic approaches to needs assessment. While it is beyond the scope of this chapter to provide detailed instructions for performing a needs assessment, a few examples are provided in the following text. For instance, the National Oceanic and Atmospheric Administration (NOAA) uses a generic 12-step process that entails the following (NOAA 2018):

1. Confirm the issue and audiences.
2. Establish the planning team.
3. Establish the goals and objectives.
4. Characterize the audience.
5. Conduct an information and literature search.
6. Select data collection methods.
7. Determine the sampling scheme.
8. Design and pilot the collection instrument.
9. Gather and report the data.
10. Analyze the data.
11. Manage the data.
12. Synthesize the data and create a report.

The following sections summarize general techniques for identifying and understanding needs in terms of strengths, weaknesses, opportunities, and threat and gap analyses. For a more extensive discussion on needs assessment approaches and models, readers are encouraged to see *Needs Assessment: An Overview* by James Altschuld (2010).

Determining Needs for Decision-Support Tools

The 12 steps identified in the previous section can be tailored for assessing and identifying decisionmaking needs among transportation system management and operations staff and agency decisionmakers. A needs assessment for a DST can be adjusted to use different methodologies and to focus on specific areas. The following is a method to perform a DST needs assessment that uses five processes to elicit needs for decision support.

- Structured interviews: Use questionnaires and interviews with operators and agencies for the documentation of current problems and issues that the DST may help solve.
- Decision analysis: Review operator actions and TMS decisions from previous incidents and events.
- Data analysis: Analyze TMS data for the identification of issues, bottlenecks, and missing functionality that could be improved through a DST.
- Technical analysis: Review computing hardware and software for the identification of issues that could be reduced or eliminated with updates to the system.
- Decision-support orientation: Provide a list of expectations from project stakeholders for the securing of resource commitment.

The reader is encouraged to see additional examples of needs assessment processes as they relate to DST implementation in other documents including Lóránt A. Tavasszy's *A DSS for Modelling Logistic Chains in Freight Transportation* (1998) and Emanuel Robinson et al.'s *Elements of Business Rules and Decision Support Systems Within Integrated Corridor Management: Understanding the Intersection of These Three Components* (2017).

Identifying Gaps and Lessons Learned

Gap analysis is an additional technique for examining needs. It identifies needs that are not being met by current processes. In other words, it is a process for comparing actual performance with desired or potential performance.

Similar to the overall topic of needs assessment, gap analysis is a broad field with a rich history of inquiry beyond the scope of this report; therefore, only a general overview of the process and suggested reading for additional information is provided.

At the highest level, the process for identifying gaps (especially with respect to DSTs) includes the following:

- Identify current processes, specifically those related to key decisions, management software, APIs, and DSTs.

- Identify current outcomes impacted by these domains.
- Compare current outcomes to desired outcomes (i.e., how decisionmaking and a DST would operate in an ideal environment).
- Document each gap in a specific and detailed manner.
- Identify processes for closing each gap.
- Focus resources on eliminating each gap (e.g., restructuring or changing processes involved in decisions or implementing DST improvements).

Table 7 recaps the discussions in this section and provides a quick reference on assessing needs for a DST.

Table 7. Summary of questions and considerations for determining decision-support needs.

Question	Area of Consideration	Consideration
How is the agency currently managing decisionmaking?	Big picture and overall motivations	• Document the current decisionmaking process. • Specify what works well and what is deficient. • Identify gaps between current decisionmaking processes and what is needed.
What are the issues the agency is trying to solve?	Big picture and overall motivations	• Determine clear performance goals and priorities. • Devote sufficient time and effort to fully define the problem before considering solutions.
What overall constraints does the agency face?	Big picture and overall motivations	• Be mindful of trade-offs, not just in DST design but also agency goals, when diverting resources to DST implementation. • Look for both obvious and subtle boundaries in the decisionmaking process; make certain that identified decisionmaking is appropriate (e.g., the TMS level). Also distinguish between decisions made by operators and managers appropriately. • Identify range of situations and accessible data.
Who are the users, and at what level will they operate the DST?	Users and expertise	Consider user backgrounds, changing workforce skills, acceptance, and feedback.

Question	Area of Consideration	Consideration
Who is going to administer the DST?	Command and control	• Have clearly designated roles been defined? • What should be the skills base and experience of the staff administering the DST? • How is the DST going to be administered, monitored, and updated?
Will there be a direct line of command and responsibility to a sole office, or will there be joint responsibility?	Command and control	Clearly outline responsibilities and priorities across various situations (if joint responsibilities).
What are the lifecycle expectations of the DST?	Lifecycle	• Be certain all elements of the lifecycle of the project, including the development process, are considered. • Consider not just immediate integration issues, but the simplicity of integrating with newer subsystems and components in the future.
Can the DST be updated modularly, or will a new system have to be installed each time?	Maintenance and updates	This question is related to lifecycle expectations and should be evaluated in coordination with the lifecycle consideration.
How much skill will be needed to update it, and who will do the updating?	Maintenance and updates	• Consider making minor changes in-house so that the system can be adaptable to constantly changing demands. • Develop maintenance and updating plans early on.
How will the system be maintained, and will resources be allocated for maintenance?	Maintenance and updates	The costs of maintenance and updates can often get overlooked (to the detriment of long-term DST viability).
Who will pay for the DST, and how will it be funded?	Funding	Investigate non-traditional avenues such as public-private partnerships; this will tie into administration and responsibility.
What DST solutions already exist?	Existing solutions and subsystems	Do not limit the search to transportation but look at DSTs in other fields.
Are there subsystem considerations in deciding on a new DST?	Existing solutions and subsystems	The best DST still needs to work well within the specific environment it is being integrated into.

Question	Area of Consideration	Consideration
Is the DST open-source or COTS?	Existing solutions and subsystems	• May combine some proprietary components and some open-source add-ins (or vice versa). • Assess any limitations on allowable functionality, the sharing of data among partners, and so on if the system contains proprietary components.
Can an older DST be modified and borrowed from for the current needs?	Existing solutions and subsystems	• Even if the entire system cannot address current needs, valuable lessons and steps can be borrowed from earlier uses of a DST.
What is the maturity of the system infrastructure that will support the implementation of the DST?	Existing infrastructure	• Even if the design process for the DST coordinates with infrastructure early on, it is imperative to determine its boundaries. It is easier to change a developing DST than to change a well-established system already in place.
What systems are in place that the DST can build from?	Existing infrastructure	• Check for existing components that can be reused in the DST or data that would already be useful to a DST as it is.
What will be the constraints on interfacing the TMS to the DST, and can they be managed so that the two can work together seamlessly?	Interface and integration	• Clearly map out potential barriers to interfacing. • Do not ignore the importance of user acceptance and usability in the DST interface (the DST may provide accurate recommendations, but if the operator does not understand them, it will not matter).
Can the DST and TMS share components, and is the DST partially integrated into the TMS?	Interface and integration	• Investigate the amount of sharing that will occur between the DST and the overall TMS, which can be both advantageous and problematic.
What are the issues and considerations for a completely integrated DST and TMS combination?	Interface and integration	• Weigh the benefits and costs of completely integrating the DST and TMS.

Question	Area of Consideration	Consideration
What are the considerations for choosing a distributed DST that is a combination of interfaced and integrated with the TMS?	Interface and integration	• Outline the pros and cons of a distributed combination DST and determine agency needs.
How will information exchange occur, especially between different agency systems?	Interface and integration	• Finding clear ways to communicate across agency systems is challenging, but try to minimize the workload for this process.
How can the impact and success of a DST be evaluated/assessed?	Evaluation and performance	• Compare the performance of the DST to similar systems and to no system at all (all-human decisionmaking). • Expect that performance will become increasingly important as DSTs become more capable of iteratively updating and adapting due to real-time feedback. Performance assessment is often overlooked in DST deployment (even outside of transportation).
How can organizational buy-in and support (especially at the most senior levels) be secured?	Business case for a DST	• Develop the case for direct benefits in performance and cost for senior management.
What is the business case for implementing a DST?	Business case for a DST	• Look beyond just economic benefits; consider overall efficiency and stakeholder satisfaction.

Question	Area of Consideration	Consideration
Are stakeholders aware of the need for a DST, and are they involved in the process early on?	Business case for a DST	• Talk with stakeholders from the beginning to understand concerns, fears, and misconceptions about what DSTs can and cannot do. Stakeholders and users are often omitted in early design planning (and then only brought in when things go wrong). • Make the business case to stakeholders also so that they are aware of the potential benefits. • Engage stakeholders early on in the planning process and encourage their ongoing involvement throughout DST development and implementation; secure stakeholder buy-in.
What are some strategies for engaging stakeholders and incorporating their needs and insights into the DST development/planning process?	Business case for a DST	• Hold discussions with stakeholders, and map out their needs and goals and the benefits of using a DST (even if deploying a DST is not what was originally planned).
What are the operational/procedural challenges that can impact successful implementation of a DST?	Business case for a DST	• Work with leadership and stakeholders to identify both internal and external sources of resistance (and potential strategies to lessen resistance). • Look for ways to fit the DST operations and recommendations within these cross-agency realities. DSTs often provide recommendations in a political/procedural vacuum (e.g., ICM). • Define objectives to consider in the context of the transportation network, traffic demand, network configuration, user/mode mixture, land use, and time of day. Based on the context, the DST should identify the most appropriate objective and performance measures.
What are the programmatic considerations?	Business case for a DST	• Ensure that the planned DST will be capable of responding to the range of operational objectives appropriate for the subject network.

Question	Area of Consideration	Consideration
What programmatic challenges may impede DST implementation and success?	Business case for a DST	• Identify areas where overall program processes may run counter to the idealized DST implementation and find room for compromise (while focusing on long-term goals).
What are workforce challenges to the design and implementation of a DST?	Business case for a DST	• Tie workforce changes and challenges back to the user assessment noted earlier, because it is essential that the user workforce buys into the system if it is to succeed.
What are potential system procurement approaches and avenues?	Business case for a DST	• Tie this topic to the funding sources sections and coordinate. • Choose contract mechanisms and approaches likely to accelerate system planning and development, optimize innovation, and minimize risk to the procuring entity or entities.

DEVELOPING SYSTEM REQUIREMENTS FOR DECISION-SUPPORT TOOLS

Requirements are the foundation for planning, designing, and building TMSs and ITSs. They determine what the system must do and are used to drive system planning, design, and development. Requirements are used to determine if a TMS's design and construction satisfy these expectations. The process of developing system requirements involves creating a requirements document for DST design and implementation. This process is vital to defining and documenting the key functions and performance requirements of a system for the agency and technical implementers of the DST. This section outlines one approach to produce a set of verifiable requirements and create a requirements document.

A good requirements development process is also essential for communicating functions and requirements down to a sufficient level of detail such that a system can be designed, developed, and implemented. The system and subsystem requirements organize and communicate what is needed to enable or establish system functionality and internal and external interfaces. These interfaces enable the sharing and use of data, constraints, performance, reliability, maintainability, availability, safety, and security. As part of the systems engineering process, a system requirements specification should be developed to provide stakeholders with an opportunity to verify that these critical aspects of the system have been captured adequately, and the project is ready to move forward to software and hardware development and implementation (i.e., build and test).

Many resources are available to assist with developing and understanding system requirements, including the following:

- The *Systems Engineering Guidebook for Intelligent Transportation Systems* (FHWA 2016) provides an overview of system requirements development and a brief template for documenting requirements. Additionally, it provides recommendations on defining needs and using those needs to develop requirements.

- *Guidance for the Development of the Set of Requirements, System Requirements Specification (SyRS)* (ISO [International Organization for Standardization]/IEC [International Electrotechnical Commission]/IEEE [Institute of Electrical and Electronics Engineers] 2011) provides a standard that can be used to develop the System Requirements Specification.

- Florida's *Statewide Systems Engineering Management Plan for Intelligent Transportation Systems* (Florida Department of Transportation [FDOT] 2005) explains requirements development and provides a template for documenting system and subsystem requirements.

In the initial stages of requirements development, stakeholders need to make certain consensus decisions to initiate and finalize system needs, which are then used to develop requirements. This process is typically difficult, especially when many stakeholders are involved. One way to begin is by developing a set of questions about proposed DST requirements and then developing answers to those questions with the consensus of the stakeholder group.

The requirements development step is important because an agency uses these requirements to communicate what the system should do to meet the needs identified by the stakeholders. Requirements serve as a reference point to verify that the system was built correctly. An agency should also establish environmental and nonfunctional requirements that define under what conditions the system is required to function to meet performance goals. Identifying system requirements can be at the discrete component level, or at a higher level that can encompass certain systems like virtual TMSs, active TMSs, temporary TMSs, and testing programs and procedures.

Following is a list of possible DST requirements-related questions from which to start. These questions may need to be revisited multiple times during the requirements development process.

- What operational decisions and tasks would one like to see performed more rapidly and with greater consistency?
- What operational decisions and tasks rely heavily on the individual operator's experience and knowledge?
- What operational decisions and tasks are so complicated or cumbersome that they are overly burdensome to some operators?
- What operational tasks would one most like to improve in their agency?
- What constraints does one have for the DST (e.g., IT requirements, existing systems that must be used, data interfaces)?
- What will be the measure that the DST is operating as expected (e.g., providing responses in a specific amount of time, providing the correct responses)?
- What outcomes are desired?
- What performance measures will be used to measure success?
- What requirements can be derived from those measures?

These questions may need to be answered for each function identified versus just answering them once for the entire system.

Types of Requirements for Decision-Support Tools

As discussed previously, requirements development is a highly iterative process. Similarly, developing requirements for a DST should begin by analyzing needs followed by developing high-level requirements to meet those needs, decomposing to system requirements, and then further decomposing down to several levels of software requirements. The requirements document should answer several basic questions, as follows:

- What is the software intended to do? What functions should it perform?

- How does the software interact with people, the system's hardware, other hardware, and other software? What interfaces does it need?

- What information and data are needed by the system? What data should the system store, use, and provide?

- What is the speed, availability, response time, and recovery time of various software functions? What is the expected performance?

- What required standards, policies, resource limits, and operations environments are in effect? What implementation language is required? What are the design constraints?

In many requirements documents, these questions are answered by classifying the requirements into various types, as follows:

- F = Functional.
- I = Interface (interface between the TMS and external systems).
- D = Data (internal storage, sending and receiving data within the TMS).
- C = Constraint.
- P = Performance.

In the field of software engineering, many other requirement types are used, depending on the methodology and expertise of the software teams developing the requirements. However, these five basic types shown should be sufficient for most TMS and DST projects.

As an example, the FDOT District 5 ICM project had more than 20 needs related to the TMS and DST, two of which are provided in table 8. After several project briefings, questionnaires, and stakeholder interviews, a set of user needs was developed by FDOT and reviewed and discussed at a workshop. After these discussions, stakeholders reached a consensus as to which needs were truly required.

Table 8. Example decision-support subsystem needs.

Number	Need	Need Description
9	Need to store pre-agreed incident response plans.	Corridor agencies need a means to collect and store pre-agreed response plans to allow them to understand collective roles and responsibilities, communicate effectively, and improve response times in reacting to events within a corridor.
16	Need to assess the impact of an enacted response plan on the transportation network.	During responses to events in their corridors, agencies need to be able to determine if preplanned responses are effective and have the intended effects. This assessment includes verifying what conditions exist after implementing responses. If system operators determine that their responses are not effective, they should be able to change components within their response plans or implement new response plans.

In table 8, user need number 9 is not a direct DST user need, but it identifies some of the functionality that the system should have to support DST functionality. System requirements should include storing pre-agreed incident response plans, which are used by DSTs to inform stakeholders which response plan to enact. This requirement implies several functions a system needs to provide to support TMS operation; for example, response plans need to be stored, accessed by users, and understood by the rules engine that will be used to select the appropriate response. Another example of a requirement that should be included is an administrative function that enables staff to add, modify, and delete plans.

Stakeholders for the FDOT ICM project wanted functionality that evaluated the impact a proposed response plan would have on the transportation network, as discussed in chapter 3; specifically, they wanted the calculate and predict and propose functions. The DST they envisioned would provide a real-time, integrated model that could predict the impact of various proposed response plans within a couple of minutes. User need 16 (see table 8) identifies this need and provides some additional requirements within the description. Note that needs are not solution-specific; therefore, while the "need to assess the impact of an enacted response plan on the transportation network" may lead some to infer that a specific technology will be used, many technologies could potentially meet this need.

When developing requirements for software-based systems, developing a well-written requirements document will do the following:

- Establish a basis for agreement between agencies and suppliers regarding what TMS and DST products will do.
- Reduce the development effort.
- Provide a basis for estimating costs and schedules.
- Provide a baseline for validation and verification.
- Provide a basis for later enhancements (especially if an agency does not have the funds to do the entire project at once).

As described in *IEEE 830*, when developing a software requirements document, there are several basic categories that should be considered by agencies, including the type of requirement, the verification method expected, and the criticality of the requirement (IEEE 1998).

Hierarchy of Requirements

Relationships among requirements should be well defined to show how the requirements are related to form a complete system. Each requirement should be uniquely identified (i.e., have a specific number or name). This identification should reflect the linkages and relationships between requirements. Agencies can show this hierarchy in many ways, but the hierarchy selected should provide a traceability back to the original need statements developed during the planning process. As an example, FDOT District 5 Regional ICM system requirements used a numbering scheme that provided requirement numbering to indicate the following hierarchy:

- Level 0—High Level System Requirements = 1.
- Level 1—TMS Software System Level = 1.X.
- Level 2—TMS Software Subsystems = 1.X.Y.
- Level 3—TMS Software Subsystem Components = 1.X.Y.Z.
- Level 4—Functions and Data Elements Within a Component = 1.X.Y.Z-A.

Once needs have been converted to high-level business requirements (Level 0 requirements), those requirements are decomposed to system-level requirements (Level 1 requirements). Agencies should carefully consider how organizing requirements is accomplished and documented so that developers will have optimal understanding of the requirement structure.

For example, in the FDOT project, the first step in the requirements development process was to translate the needs, develop system-level business requirements, and then decompose the business requirements to subsystem-level software requirements. The software requirements were numbered based on level of importance within the system and included several types, as follows: functional, performance, interface, data, and hardware. They also covered nonfunctional and enabling requirements and constraints. In table 9, the requirement identifier provides the level within the system for the requirement.

Table 9. Example system-level requirements for an integrated corridor management system.

Requirement Number	Requirement Description	Type	Need	Verification	Critical	System
1.9	The ICM system will store pre-agreed incident response plans.	D	9	Demonstrate	H	DSS
1.16	The ICM system will evaluate the impact of enacted response plans on the transportation network.	F	16	Demonstrate	H	DSS

D = data; F = functional; H = high.

In table 10, the software requirements are further decomposed to subsystem-level requirements. These provide an indication of which subsystem they are a part of—in this case, the DST—and the method for testing that will be used to verify that the requirement is met during the testing phase.

Table 10. Example subsystem requirements.

Requirement Number	Requirement Text	Type	Parent Requirement	System	Verification
1.9.1	The DSS shall store pre-agreed incident response plans as defined in the data dictionary.	F	1.9	DSS	Demonstrate
1.9.2	The DSS shall provide the ICM manager the capability to add pre-agreed incident response plans for a specified incident.	F	1.9	DSS	Demonstrate
1.9.3	The DSS shall provide the ICM manager the capability to query pre-agreed incident response plans.	F	1.9	DSS	Demonstrate
1.9.4	The DSS shall provide the ICM manager the capability to edit pre-agreed incident response plans for a specified incident.	F	1.9	DSS	Demonstrate
1.9.5	The DSS shall provide the ICM manager the capability to delete pre-agreed incident response plans for specified events.	F	1.9	DSS	Demonstrate

F = functional.

In table 11, the expert rules engine (ERE) illustrates how DST requirement 1.9.1 (see table 10) is further decomposed into requirements for the ERE component.

Table 11. Example expert rules engine requirements.

Requirement Number	Requirement Text	Type	Parent Requirement	System	Verification
1.9.1.1	The ERE shall store pre-agreed response plans in a network-accessible location.	F	1.9.1	ERE	Demonstrate
1.9.1.2	The ERE shall provide the prediction engine with pre-agreed response plans.	F	1.9.1	ERE	Demonstrate
1.9.1.3	The ERE shall provide the evaluation engine with pre-agreed response plans.	F	1.9.1	ERE	Demonstrate

F = functional.

This process illustrates the importance of iteration—moving from needs to requirements to increasingly detailed requirements and repeatedly consulting with stakeholders and the technical team at every stage of the process.

Verifying and Validating Requirements

Software verification and validation help an agency determine whether the software requirements are implemented correctly and if they are meeting needs as intended. Within the requirements document, each requirement should indicate how the agency expects the requirement to be verified and validated. The types of verification and validation typically seen in requirements documents for TMSs are as follows:

- Analyze: Use established technical or mathematical models or simulations, algorithms, or other scientific principles and procedures to provide evidence that an item meets its stated requirements.

- Inspect: Observe with one or more of the five senses simple physical manipulation and mechanical and electrical gauging and measurement to verify that an item conforms to its specified requirements.

- Demonstrate: Operate an item to provide evidence that it accomplishes the required functions under specific scenarios.

- Test: Apply scientific principles and procedures to determine the properties or functional capabilities of an item.

Priority or Criticality of Requirements

Agencies should identify the priority or criticality of each requirement during the consensus-building process that is part of planning and designing the TMS. As appropriate, a scale—such as rating importance on a scale of 1 to 10 or a simpler scheme—can be used to identify the priority of each requirement. In the majority of TMS and DST projects reviewed, a simpler scheme was used in the requirements document, which was typically as follows:

- H = high.
- M = medium.
- L = low.

Finalizing Requirements and Beginning the Design Phase

There are many ways that requirements documents for the design and implementation of a DST can be developed; the example above is one approach. Agencies need to write well-defined requirements and organize them based on the specific needs identified during the planning phase. A DST system requirements document has two primary audiences—the agency and the technical implementers of the DST—and it serves to document an agreement between them.

CHAPTER 5. DESIGNING, DEVELOPING, AND DEPLOYING DECISION SUPPORT TOOLS FOR TRAFFIC MANAGEMENT SYSTEMS

This chapter introduces the processes of designing, developing, and deploying a DST and describes the following stages of the systems engineering process:

- High-level design.
- Detailed design.
- Development.
- Verification.
- System validation and deployment.

This chapter addresses the common issues and potential implications of decisionmaking (e.g., selecting technologies) that arise in each of these stages, presents lessons learned from agencies who have coped with those issues, and considers the following development scenarios:

- A stand-alone DST is deployed and shares information with an existing TMS and TMC.
- A DST is integrated with an existing TMS.
- A DST is deployed as part of a new TMS.

The objectives of this chapter are as follows:

- To provide an overview of the key stages of the systems engineering process and their applications to design, develop or select, and integrate a DST into a TMS.
- To identify options for the modification or integration of an existing DST to improve the operation or real-time decisionmaking of a TMS.
- To identify issues for consideration while ensuring to address operational objectives and requirements throughout the design, development, and selection processes.

This chapter introduces methods for using an agency's needs and requirements to assess technologies, design a DST, and integrate it into the plans for and design of a new or existing TMS. Additionally, it focuses on how to use these requirements in the selection, design, and development processes.

HIGH-LEVEL DESIGN

This design phase begins the transition from user needs and requirements to design so that developers can begin the development process. At the conclusion of the design phase, there should be a clear picture of architecture, data elements, interfaces, electronic messages and protocols, UI specifications, software changes or API specifications, and user interactions.

The deliverables for this design phase usually include the following:

- Architecture: Shows necessary logical systems and interfaces and standards to be used for the interfaces.

- Preliminary UI: Shows some preliminary descriptions and pictures of the potential UI.

Data Subsystem Design

As discussed in chapter 3, the data subsystem consists primarily of the data and the interfaces or APIs that extract data from other systems and devices to enable the data to be transformed and loaded into the TMS database. The data subsystem should be identified early in the process, because availability of certain data may constrain the types of analytics or process automation that are possible within the system. When designing a DST, agencies and developers need to ask the following key questions:

- What data does the DST need to provide the decision support needed?
- Do the data currently exist in the systems? If not, is there a way to get the missing data?
- Are these sources for new data elements already integrated into the TMS data subsystem, or will they need to be integrated with the TMS? Will the new data elements also need to be added to the data subsystem?
- Do the new sources of data have defined APIs that can be used to integrate these new sources into the TMS and/or data subsystem?
- Do the current data provide the requirements needed (e.g., 10-s intervals instead of 60-s intervals, area of coverage, accuracy)?
- Is the quality of the available data sufficient to support decisionmaking?
- Is the coverage area of the data sufficient to support decisionmaking?

The data subsystem should store data that the DST needs to operate and to facilitate data exchange with other systems. It also needs to provide the administrative and maintenance interfaces necessary to manage data access and monitor the system. The following section discusses aspects the design of the data subsystem should address.

Data Types

Within the transportation domain, common measurements and observations are used by many different agencies. Understanding the range of common data elements needed by the system will help to scope the requirements and effort. Examples of data types include traffic data (e.g., volume, density, and speed) and incident data (e.g., duration and severity). For data to be effective in a DST, they must be stored using common formats and interpretations. If they are not, the data subsystem and DST may require reformatting, or data may have to be transformed as they are collected from various sources before they can be used by the data subsystem or DST.

Data Formats

Once the necessary data types are well understood, system developers should consider which, if any, data format standards to use. Examples of data format standards include the *Traffic*

Management Data Dictionary (Institute of Transportation Engineers 2020) and the *IEEE 1512* (IEEE 2006). These decisions may impact the data storage approach but will be particularly important in establishing data import and export requirements.

Data Validation and Cleaning

Different data sources may have different formats, data quality, or other distinguishing attributes. Platform designers should consider what data conditions are valid (i.e., which values may or may not be missing), what data ranges are acceptable, and what a system does in response to invalid data.

Storage Approach

When data are accumulated, they can be stored in raw form as collected, aggregated, or otherwise restructured. The use cases for accessing data as they age will determine the best long-term storage formats. Aggregating data can reduce storage space and speed up some operations, but the system may lose some capability for fine-grained analysis. Designers may decide to store raw data for a certain period of time and then aggregate the data for longer term storage.

Archival Requirements

Once aspects of data access and storage are determined, designers need to decide how long data will need to be retained and, if multiple storage approaches are being used, in what formats.

Data Volume

Sizes of data elements, frequencies of exchanges, and desired retention periods should all be considered to ensure the network and communication infrastructure support a regional DST needs. Communication capacity considerations should take into account short-term burst rates for data flow, and storage capacity should be based on long-term data flow rates.

Data Management Plan

The data subsystem requires routine and preventive maintenance and management, including processes for server hygiene and data migration and retention.

Selection of Decision-Support Tool Type

Once the high-level design of the data platform has been completed and related fundamental constraints identified, the next step is for designers to make a final decision on the type of DST that will meet the needs and requirements of a TMS.

Availability of certain data types will influence the design and selection of the DST. If data are not available or are not available at the level of detail needed, the DST selection should take that into consideration. For example, many TMS data subsystems provide roadway link data that are averaged over a period of time and across lanes. If the DST needs to use individual lane data, changes to the TMS may be required.

At this stage, system designers should make a final decision on the type of DST to be deployed. The various types of DSTs, which were discussed in more detail in chapter 3, are as follows:

- Knowledge-Driven DST—Plan selection: Selects between a set of predefined response plans based on current conditions.
- Knowledge-Driven DST—Rules engine: Applies a set of business rules to the current conditions and generates response plans.
- Data-Driven DST—Algorithm engine: Applies one or more deterministic algorithms to current conditions to generate response plans, detect events, and so on.
- Model-Driven DST—Simulation engine: Executes a simulation model to assist in evaluating various response plans and predicting what near-term conditions may be for each response plan.
- Data-Driven DST—Machine learning engine: Uses artificial intelligence or other methods to evaluate conditions; assess different responses; and select operational strategies, control plans, and other responses or actions to implement.

As noted directly above, chapter 3 provides a more detailed description of the different types of DSTs, along with a brief discussion of their advantages and disadvantages and when various types may be applicable to consider using. However, a DST may also use a combination of these different types. A common motivation for using multiple types of DSTs is to satisfy different operational needs with responding to or completing varying tasks or stages associated with different events. For example, an agency may want to use an algorithm engine for event detection, a rules engine for response generation, a rules engine for ramp management and control, and an entirely different DST which could be standalone software program to identify and predict weather conditions.

Based on the data availability, computing requirements, and type(s) of DST desired, an analysis may be needed to ensure the DST will be able to provide the specific decisions using or translating the data required from the data subsystem and for other TMS subsystems or systems to implement decisions.

Workflow Design

The use cases identified in the ConOps and the various scenarios within those use cases need to be articulated into clear workflows for the system operators. These workflows should clearly delineate between automated tasks performed by DSTs and manual tasks performed by operators. It may be useful to express workflows with flowcharts or Unified Modeling Language, as these visually oriented tools help to make workflows accessible to nontechnical audiences. Workflows should be validated with operational staff and against real-life examples of the problems DSTs are expected to address.

User Interface Design

To the degree possible, DSTs should maintain UI conventions and approaches that are familiar to operators. Having a similar look and feel between a TMS and its DST will simplify the rollout process and may increase user satisfaction with the new system. If the DST is an entirely different system from the TMS, this may be more challenging than if the DST is a new module within an existing TMS. UI designers should have access to systems currently used by operators so that they can understand user expectations and preferences.

The design process should include a phase to develop wireframes, which are layouts of user screens that identify the major interface elements but do not specify details of look and feel. Wireframes are the transition point from workflows to visual design. Operators and other stakeholders can use wireframes to confirm that workflows are correct and effectively presented.

Response Design

It is critical that the technical implementation of the response generation process does not inadvertently diverge from the operational goals set forth in the ConOps. Such a divergence can occur when an aspect of a response gets lost in translation between the ConOps and technical implementation.

Designing the encoding of response rules and algorithms should be an iterative process involving both technical and operational staff. In the case of a rules engine, this iterative process will involve having technical staff convert the response rules described in the ConOps into formal business rules that the system can use. Technical and operational staff will need to work together to confirm that the formal business rule definitions match the intent of the ConOps. Walking through real-life examples with both technical and operational staff is a good approach to validation.

Computing Hardware Design

The requirements and design for a computing hardware to support a DST should follow IT standards and TMS requirements. If a legacy TMS exists, agencies should evaluate whether the current computing hardware will support the added DST requirements and software or if additional computing capabilities or a new platform will be needed. When developing the requirements and design for a DST's hardware platform, agencies and developers need to ask the following key questions:

- Does the DST require hosting within the agency's TMS software platform or subsystem?
- Does the current hosting environment have the available resources (e.g., processing, memory, and hard disk space) needed to support DST requirements?
- Can additional capabilities or resources be appended to the existing hardware platform?
- Does the current hardware platform need to be replaced?

- Can changes be made to the hardware platform given the software being used for the TMS?

The Dallas ICM DST was a multiagency platform that could not be hosted on the existing computing hardware of the agencies. Their existing TMS was fully using their computing hardware; therefore, new computing hardware specifically for the ICM system and associated DST was designed and deployed. The FDOT District 5 TMS uses a large, set-computing hardware for their TMS, and they designed future expansion into its original computing hardware. After evaluating the data, processing, storage, and memory requirements of their new DST, FDOT and the software development vendor determined that the existing TMS hardware platform or subsystem could support the additional software needed for the DST.

DETAILED DESIGN

In this stage of designing a DST, the design document is refined and finalized so that software programmers have the information they need to begin writing code. A common challenge in DST development is that, when a rule is formalized into something that can be incorporated into an algorithm or software program, the original intent of the rule can be lost. To address this risk, operations staff need to apply extra vigilance and work closely with technical staff to ensure the formal definition returns the correct results.

The deliverables for this phase of design usually include the following:

- Data interface specifications: Describe each data interface, the relevant standards that will be used (e.g., National Transportation Communications for Intelligent Transportation System Protocol), how the data will be exchanged (i.e., format and frequency), and the content of the data. An interface control document is usually created by the software developer to document the data interface and is used to create the API to facilitate the exchange of data between each system and subsystem within the TMS.

- Detailed UI: Provides a more detailed UI description along with images of the UI.

- Detailed rules/algorithm/analysis specifications: Describe the rules that the DST will use, the ranges of expected values for each rule, and the formulas or algorithms that the DST will use to calculate various measures.

- Detailed functional descriptions for each component: Describe each component of the DST, how it will interface to other components, the data it requires from other systems, and the functionality the component provides in relation to requirements developed during the planning phase.

- Critical design review: Schedule a meeting between the design team and project stakeholders to review the almost-completed system design; this meeting usually occurs when 90 percent of the design is complete.

The detailed design phase takes the concepts and preliminary design developed in the high-level design phase and makes the technical decision on what will be developed, provides more detail, and documents the complete specification for each of the DST components. Questions provided in the previous sections for the various components should be revisited by the designer and further detail developed. The type of DST being developed is finalized at this point. While the high-level design provides a logical and physical architecture at a conceptual level, the detailed design should include the following:

- Functional logic of each DST component: The logic, algorithms, and processes for each component should be finalized and documented to allow a software developer to configure or develop the software components of the DST.
- Database tables finalized: The data subsystem design should be finalized and documented to include the type and size of all data elements and a data catalog that identifies all data within the data subsystem.
- Complete detail of the interfaces: The APIs and the associated interface control documents should be developed and finalized to allow developers to integrate the various TMS software subsystems with the DST.

The detailed design phase should provide a complete design that can then be used by software developers and system integrators to develop the DST and integrate it into the TMS.

DEVELOPMENT

The development process is mostly managed by the technical team, and most issues that determine its success or failure are not unique to DST development. Project leaders should maintain good visibility into the progress of the project. In the case of an agile process, taking this approach means taking part in sprint review meetings regularly. If using a waterfall approach, then managers should schedule multiple intermediate demonstrations. At sprint reviews or intermediate demonstrations, project leaders need to ensure that the system behavior has not inadvertently diverged from the operational goals.

In the case of a DST, taking this approach means checking that operational use cases for incident and congestion response are still valid. The development process (agile, waterfall, etc.) will depend on factors such as the requirements of the agency, the complexity of the project, and risks.

The deliverable for this phase is a working DST, which can be developed through either an iterative (e.g., agile) or sequential (e.g., waterfall) development processes.

VERIFICATION

Verification is an ongoing process that involves testing each subsystem as it is assembled and integrated to ensure its performance meets design requirements. This process is repeated at each level of integration, until the entire system is assembled, at which point the entire system's performance should be verified one final time. Verification entails confirming that the system being developed fulfils its design requirements (all requirements identified in the design stage) and is usually done in a laboratory or factory environment. Verification is typically conducted using an installation of the system dedicated to development or quality assurance (not under real-world conditions). Most of this process falls within typical software development procedures, and project managers should ensure the project adheres to best practices.

However, there are some aspects of verification that are particularly challenging for DSTs. DST behavior often depends on a complex combination of inputs from multiple data sources. It can be difficult to recreate these large and varied input streams in a way that is useful for repeatable testing and verification. Some approaches include the following:

- Randomized inputs: Testing tools can be created to provide random numbers that approximate data sources (e.g., traffic detector readings). These tests can be made repeatable by reusing the same random number seed. This method is a simple way to produce data that can be input to the DST for testing. A drawback to this approach is that the data may not match realistic inputs when viewed from a system perspective. A key aspect to a successful DST is that it responds correctly to conditions over a large area or time interval, and it is very difficult to get a coherent data set over wide-ranging values of time and space using randomized inputs.

- Record and playback: Testing tools can be built that can record live data inputs and then play them back for testing purposes. This process is a relatively simple way to collect data that is realistic and coherent over time and space. A limitation to this approach is that collecting data reflecting extreme circumstances may be challenging.

- Traffic simulation: A realistic traffic simulation can be created and then used to derive inputs such as traffic volumes and speeds. This approach is a good way to create realistic test scenarios, and simulation parameters can be varied to create extreme conditions. A drawback to this approach is that can it be very expensive and time consuming to build and calibrate a model if one does not already exist.

The deliverables for this phase usually include the following:

- Test plan: Provides a step-by-step script for an agency to follow and verify that the system developed meets the requirements determined during the planning and design phases.

- Test readiness review: Requires a meeting between the development team and stakeholders to review the results of internal testing and ensure the system is ready for system acceptance testing by the agency.

- System acceptance testing: Requires verification of the completed system to ensure it meets all requirements and performs as expected and the agency accepts the completed system and is ready to deploy and begin operations.

SYSTEM VALIDATION AND DEPLOYMENT

The process of installing the system in the field, confirming that it functions in its intended environment (performing according to system requirements), and confirming it meets the intended user needs constitutes deployment and validation. This process can be challenging for any TMS. The simplest approach is to switch operations over to the new system, but this approach has the potential to disrupt operations. This problem is particularly difficult with DSTs, because the situations necessary for DST validation (congestion and incidents) are the situations where it is the most critical that operations are not disrupted. It is ideal if the new and old systems can run in parallel for a period of time. This practice allows operations to continue uninterrupted in addition to observing the behavior of the new system under field conditions.

The deliverables for this phase include the following:

- Validation report: Documents any issues or nonconformances to ensure that all requirements and use cases are validated.

- Change request: Discovers any additional requirements or desired changes to require a change request to the DST developer through the verification and validation process.

- Operation of the DST: Prepare the DST for use once system acceptance has been completed and approved by the agency.

CHAPTER 6. MONITORING, EVALUATING, AND UPDATING A DECISION-SUPPORT TOOL

This chapter provides an overview of the processes for monitoring, evaluating, and reporting on the performance of a DST once it has become operational. It also addresses how agencies may use the results of an evaluation to consider when enhancements may be appropriate to improve how it is being used or operating or when a DST may no longer be needed. The objectives of this chapter are to describe the following:

- Measuring DST performance and collecting data.
- Monitoring DST performance.
- Evaluating DST modification of operation, enhancement, or replacement for appropriateness.

After reading this chapter, one should understand some of the performance measures, techniques, and methods that agencies use to evaluate and ensure their DST is operating as expected and providing satisfactory results. One should also understand how to determine when changes are needed to the DST and TMS or TMC.

MONITORING AND EVALUATING DECISION-SUPPORT TOOLS

As discussed in previous chapters, a DST provides the critical information and decisionmaking support necessary for transportation operating agencies to understand the increasing volume of incoming data and decide among a complex array of alternative actions. The literature available discusses many processes and measures for evaluating the performance of a TMS. However, very little information is available about monitoring and evaluating the operational or decisionmaking performance of a DST.

Assessment and Evaluation Methods

Assessing DST performance is highly dependent on the type of DST and the data available from the TMS, outputs of the DST (its functions and types of decisions), and data from various field devices. From a computer system point of view, many tools and processes are available to ensure that the hardware, network, and software are operating correctly. However, from an operational perspective, agencies will also need to monitor various elements of the system (both the TMS and the DST) to evaluate whether the DST is operating correctly, including the following:

- The availability and accuracy of data, information, or content as decision inputs.
- The presence of any decision-process bottlenecks and traffic (i.e., whether decision calculations are taking too long due to a resource constraint within the system).

The existing literature for DST evaluation is domain-specific; however, in general, performance measures can be established through the assessment of four different characteristics: effectiveness, efficiency, use, and satisfaction. Some example measures that address these items follow:

- The correctness and precision of the response plan provided by the DST: Is the recommendation accurate?

- The percentage of problems resolved following a decision: Do the responses recommended improve network performance?

- The speed of decisionmaking: Does the DST take too long to recommend a response?

- The time it takes a TMS operator to accept/reject/modify a recommended response: Is the process for the operator efficient and timely?

- The productivity of the DST: How many responses were calculated for a given event?

- The operator's confidence in the predictions and responses generated by the DST: Does the operator trust the system? (This is a qualitative measure.)

During evaluation, an agency should consider the purpose of the DST and how to measure whether it is performing correctly. DST performance should be both accurate and timely to facilitate operator decisionmaking. To develop evaluation criteria, agencies can reference the following resources:

- *TMC Performance Monitoring, Evaluation and Reporting Handbook* (FHWA 2005). Available online at: https://tmcpfs.ops.fhwa.dot.gov/projects/tmcpmerhdbk.htm.

- *Evaluating Alternative Operational Strategies to Improve Travel Time Reliability* (TRB 2013). Available online at: http://www.trb.org/Main/Blurbs/168142.aspx.

- *Guide to Effective Freeway Performance Measurement: Final Report & Guidebook*, (Margiotta et al. 2007). Available online at: https://transportationops.org/publications/guide-effective-freeway-performance-measurement-final-report-and-guidebook-0.

- *Traffic Analysis Toolbox Volume I: Traffic Analysis Tools Primer* (Alexiadis, Jeannotte, and Chandra 2004).

- *Traffic Analysis Toolbox Volume VI: Definition, Interpretation, and Calculation of Traffic Analysis Tools Measures of Effectiveness* (Dowling 2007). Available online at: https://ops.fhwa.dot.gov/trafficanalysistools/tat_vol2/Vol2_Methodology.pdf.

As discussed in the *Guide to Effective Freeway Performance Measurement: Final Report & Guidebook* (Margiotta et al. 2007), developing a performance measurement program that can be used to evaluate TMS and DST effectiveness begins with identifying the data available to perform the evaluation. The following sections outline recommended steps for developing performance measures to evaluate DST and TMS performance.

Select Measures for Performance and Effectiveness

Based on the DST selected, performance measures and measures of effectiveness associated with the evaluation should be determined to address the four different characteristics—effectiveness, efficiency, use, and satisfaction—used for TMSs and DSTs.

Obtain Existing Data

Because most available performance data are collected for purposes other than reporting performance, obtaining data from existing traffic management and traveler information systems is often far more difficult than logic would indicate. Budgets for developing and deploying traffic management and traveler information systems are often smaller than needed; a very common cost reduction strategy is to remove or significantly curtail the data-archiving function required to store and efficiently retrieve TMS data. Consequently, agencies must often start creating a performance monitoring system by constructing the software necessary to efficiently store and retrieve data that is already being collected by an existing TMS.

Develop Procedures for Data Manipulation

Once obtained from the data subsystem, the data collected from the TMS can do the following:

- Be used directly as a measure of performance (e.g., vehicle volumes, vehicle speeds).
- Be combined with other data from other devices in the field to compute a new performance statistic (e.g., travel times computed from toll tag readers at different locations).
- Be mathematically transformed into a different performance measure (e.g., point speeds at consecutive locations can be used to estimate travel times along a corridor).
- Be combined with other directly measured data to produce more complex performance measures (e.g., vehicle volume and speed data can be used to compute vehicle hours of travel or vehicle hours of delay).
- Be used as input to various transportation modeling systems to estimate a wide variety of statistics that cannot be measured by the available sensor equipment (e.g., volume data used to feed a simulation model can produce estimates of pollution emissions).

Seek To Understand Available Data

Once all the available data have been identified, agencies should seek to clearly understand exactly how well those data represent the performance of the transportation route they are being collected on. Agencies should also investigate how limitations in the data affect their use as performance measures. Understanding the strengths, weaknesses, and holes in the available data helps determine what supplemental data need to be collected specifically for performance monitoring. It also defines the need for many of the data manipulation steps and many of the assumptions that must be made to convert the available data into the required performance statistics.

Collect Supplemental Data

Once a clear understanding of the available data exists, it is possible to define the supplemental data collection that is needed to complete the necessary data sets for the desired performance monitoring system. Supplemental data will be used for the following purposes:

- To fill in the gaps in available data.
- To provide information that helps eliminate biases in the previously collected data.

Obtain Performance Measures and Develop Reports

Using the selected performance measures and measures of effectiveness and the data collected, performance reports should be developed to show how the DST is performing based on the effectiveness, efficiency, use, and satisfaction recommended to evaluate the performance of a DST.

Develop Corrective Actions

If a DST or TMS is not performing as desired (or expected), some corrective actions may be needed. These actions could include modifying processes and procedures used by the operators of the TMS or identifying changes needed to the DST. Maintaining TMSs and DSTs requires monitoring and evaluating the performance of the system on a regular basis. This process is discussed further in the sections below on updating a DST.

Decision-Support Tool Evaluation Example

For the ICM program, the USDOT evaluation team developed an evaluation plan for both the Dallas and San Diego ICM projects. These evaluations considered several key performance and accuracy measures, as follows:

- The quality of responses generated by the DSTs.

- The accuracy of the simulation software's predictions as to transportation system conditions 30 min or more into the future.

- The speed of response plan generation.

- The varying conditions and data loads (e.g., minor versus major incidents) that impact TMS and DST performance (i.e., how quickly a decision could be calculated based on the load on the computing platform and data platform).

The evaluation of the DST considered different measures of effectiveness to be analyzed, including the following:

- Percentage of times a TMS operator implements responses recommended by the DST.

- Percentage of times a TMS operator alters recommended responses (without dismissing them completely).

- Average time for the DST to deliver an actionable response plan.

- Average time for the DST to deliver predictions of response plan effectiveness.

- Average number of response plans generated per event hour.

In addition, the following qualitative measures of effectiveness were selected and used as a basis for the evaluation:

- Responses consistent with the operator's experience and perceptions (per the TMS operators).

- Perceived quality of responses, including improvements relative to any comparable pre-ICM approaches (per the TMS operators).

For the evaluation, the data were collected through a mixture of interviews directly from the TMS data subsystem by the evaluator. The performance measures were developed offline after the project was completed.

Through the APIs provided by the Dallas ICM TMS, the evaluator connected to the data subsystem and routinely downloaded all data into their own data platform used for the evaluation.

UPDATING THE DECISION-SUPPORT TOOL

A DST may require an update to its hardware, software, logic, or underlying data when it is no longer performing as expected, new information is available, or changes have been made to one of the components. Updates are also required when the computing platform or subsystem, the DST (e.g., software and algorithms), or the TMS has reached the end of its life or upgrades may be made to one component. When changes occur in the TMS, agencies should evaluate and plan for potential changes to the DST. Depending on the complexity of the change, the enhancements to the DST or APIs could be done simultaneously or after the TMS changes have been made. If there are numerous changes, agencies might want to consider doing one subsystem at a time so that if errors occur, they are more easily determined.

All TMS and DST changes should follow a change management process, with defined back-out procedures in case the changes do not deliver the expected results or the change fails. According to IT Infrastructure Library process framework found on the Information Technology Infrastructure Library (ITIL) Open Guide website (nd), the goal of change management is to control risk and minimize disruption to associated IT services and business operations.

Computing Hardware Updates

Computing hardware or subsystem updates usually occur when the current hosting servers are near or at end of life. Due to growth, budget cuts, rack limitations, and other factors, servers deployed for one purpose must often begin fulfilling additional services and responsibilities; therefore, it is important to audit systems periodically. Reviewing a server's resource load helps ensure an organization optimizes performance and prevents downtime. Server upgrades always require planning.

Software and API Updates

Like the process for updating hardware, the processes for updating server operating systems, tools, APIs, and database and other software should be thoroughly tested. Additionally, a back-out procedure should be developed in the event an upgrade does not go as planned.

Rule and Algorithm Updates

The newer generations of DSTs are beginning to have self-evaluating capabilities. Staff can support this process by monitoring and evaluating the effectiveness of decisions made and the utilization rates of DSTs. If cases arise in which recommended decisions run counter to preset agreements and boundaries in a DST's logic, then a DST system should be reviewed for areas of improvement. Depending on the type of DST used, the business rules or predictive model may require updates based on changes in the data or analysis of a DST's operation.

Data Updates

Changes in the transportation network can occur frequently, depending on where the data originate (e.g., third-party data, detectors, and other TMSs). For the majority of TMSs, two types of data changes may occur: static and dynamic. Changes in static data (e.g., roadway network and device location), as the data's name suggests, do not occur often. Changes in dynamic data (e.g., speed, volume, and occupancy) are more frequent, as these types of data are constantly changing. The accuracy of static data used by a TMS can significantly impact the operation of a DST. A defined update period is recommended to ensure that changes can be made to TMS subsystems, APIs, or components that may be affected by data changes.

Additionally, new sources and types of data may become available as a TMS matures (e.g., data from additional field detectors, third parties, and connected vehicles). When these sources become available, an analysis should be done to determine if these data will improve a DST's accuracy and reliability. Just because a new data source is available does not necessarily mean it should be integrated into a TMS and used by a DST.

For example, the Dallas ICM TMS automatically extracted and shared data via external and internal interfaces and served as the central data store for the "system of systems." The following data updates within the Dallas ICM TMS were performed on a quarterly basis.

- Roadway network updates (new road names, new roads, changes to speed limits, etc.).
- Transit route and schedule updates (changes to bus routes, bus schedules, etc.).

- Response plan updates, including the following:
    - Incident response plan data, including response plan identifiers and plan descriptions.
    - Incident response plan dialog data, including response plan request identifiers, agency identifiers, and decisions.
    - Plan decision results data, including response plan identifiers and decisions.

The following real-time data were managed by the systems maintenance team for the Dallas ICM DST.

- Real-time and static network status data, including the following:
    - Incidents, both traffic and transit events.
    - Construction and planned events, including schedules.
    - Dynamic data with links, including speed, volume, and travel time data for roadway links.
    - Traffic signal status data, including the operational status of traffic signals.
    - DMS status data, including the operational status of signs and text representations as available.
    - High-occupancy vehicle status data, including the operational status of high-occupancy vehicles as available.
- Parking management system data, including transit park-and-ride lot utilization and parking space availability.
- Transit data, including the following:
    - Transit route.
    - Schedule adherence information.
    - Current location of Dallas Area Rapid Transit buses and light rail vehicles.
    - Transit incidents.
    - Passenger load of the red and orange light rail lines.
    - Construction and special event information.
- Weather data, including weather alerts.

These data provide a robust basis for a network-based, multimodal evaluation of the project and for establishing and monitoring the corridor operational performance measures used to evaluate the DST's effectiveness. Ensuring that these data feeds and interfaces are up to date is a requirement for any DST to continue to be accurate and effective. If the data are out of date, the recommendations made by the DST will be inaccurate and could potentially worsen traffic network performance.

It can be expected that many of these data sources have changed and will continue to change over the life of the project. These changes may be due to new services being implemented, roadways being built or renamed, response plans being updated, new response plans being developed and implemented, and additional ITS field equipment (DMSs and CCTV cameras) being deployed. Ongoing maintenance and update activities can be expected to incorporate adjustments into the DST based on these changes in the built environment and on the types of new information and devices that will become available over time.

CHAPTER 7. EXAMPLES OF DECISION-SUPPORT TOOLS FOR TRAFFIC MANAGEMENT SYSTEMS

PURPOSE AND OBJECTIVES

This chapter introduces several examples of DSTs that have been developed and deployed. As discussed in the earlier chapters of this report, there are many tools that can be used for decision support and many tools that can operate as elements of a DST that are fully integrated into the real-time operation of a TMS and TMC.

The objective of this chapter is to provide case study examples for each of the three types of DSTs discussed previously. These case studies provide an overview of several projects that have included developing and deploying DSTs at various phases in the lifecycle of the TMS. A scan of current practices led to the conclusion that there are few DSTs in the United States that are used in the real-time operation of a TMS; however, many DSTs are used in TMSs to assist operators with decisionmaking.

Most existing DSTs identified were associated with TMSs that manage and control traffic on freeways. The purest forms of DSTs are used for device control (typically motorist information systems such as DMSs and HAR) and assisting operators with specific action recommendations in the operations center or workstation of a TMS.

Some systems did not have automated DST capabilities but had procedures in place to perform manual decision-support activities. Such an implementation of procedures is a positive indicator that a future DST, implemented using sound systems-engineering processes, will be successful. The DSTs examined in the scan of current practices were limited to providing the following services:

- Dissemination of en route traveler information using ITS devices such as DMS location selection for activation, DMS message library selection, or dynamic message generation.
- HAR station selection and message generation.
- Operator-recommended actions (e.g., contact service patrols and emergency responders).
- Key personnel notification (e.g., e-mail, phone, text message).
- Traveler information dissemination (e.g., website and interactive voice responsive).

- Transit route adjustments or action plans to address service interruptions.
- Treatment plans to mitigate the impact of winter weather on road conditions.
- Dynamic pricing for managed lanes.
- Active TMSs.
- Weather prediction.

CASE STUDY: KNOWLEDGE-DRIVEN DECISION-SUPPORT TOOL—MARYLAND COORDINATED HIGHWAYS ACTION RESPONSE TEAM

The Maryland CHART TMS helps operators manage traffic events in the State of Maryland. The DST element of the TMS determines the best DMS, HAR, and CCTV camera devices to use in response to a traffic event and suggests messages for the operator to consider putting on the selected DMS and HAR devices.

As discussed in chapter 2, a knowledge-based DST provides specialized problem-solving expertise based on the processing of stored facts, rules, procedures, or similar forms of knowledge. They attempt to emulate human reasoning, but with more consistent results. Based on the CHART Business Architecture plan, the primary reason for adding the DST to the TMS was a need to "improve the management of traffic" by improving travel time reliability, reducing congestion, and several other measures of effectiveness. The CHART operators use the DST functions to initiate responses to events that will optimize traffic flow on roadways, clear incidents, and re-open lanes as quickly as possible, while also protecting the safety of victims, travelers, and emergency response personnel.

The CHART DST consists of a rules engine and set of rules that use network travel patterns (driving directions) from each device to a particular traffic event to determine which devices may be useful for the event. All devices that are upstream from the event are considered, even those not on the same primary route as the event. If the driving directions from the device to the event are within a certain small number of turns and within a certain very small number of U-turns, the device will be considered. The driving directions from each device to the event can also be displayed on a map, aiding the operator in making the final determination as to whether the device is appropriate to use for that event.

The Process of Decision Support

Once an incident is verified, the operator can request the DST recommend response plan elements (e.g., DMS messages and HAR messages to disseminate and CCTV camera footage to review). This information is provided to the operator through a prioritized list based on a scoring algorithm, and the operator decides which actions to implement. To support the dissemination of DMS and HAR messages, the DST provides message templates that are automatically generated based on event information, but the operator decides what changes are made to the messages prior to posting.

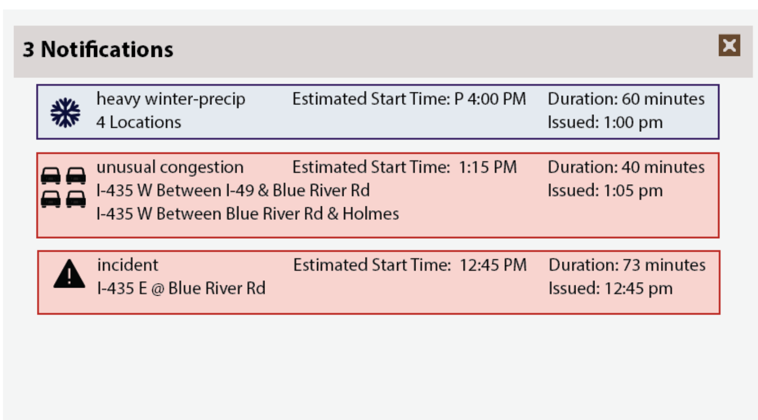
The CHART DST is configured to suggest 511 messages for an individual event as part of a traffic event response plan. The user decides whether to post the message to the Maryland 511 system and can override generated text.

Although the decision-support suggestions revolve around traffic events, it is not difficult to imagine the use of decision-support-like suggestions for other portions of the CHART active TMS. To ensure this type of flexibility, the decision-support-enabled interface has been defined outside of the traffic-event domain. This Common Object Request Broker Architecture interface can be implemented by any element of the TMS that could use recommendations from the DST.

CASE STUDY: HYBRID MODEL-DRIVEN AND DATA-DRIVEN DECISION-SUPPORT TOOL—KANSAS CITY SCOUT ELECTRONIC TRAFFIC MANAGEMENT SYSTEM

Data-driven DST use data to aid in the decisionmaking process. They typically use data from databases that are designed to be queried, which enables processing and analysis of the data to develop insights that support decisionmaking. The effectiveness of a data-driven DST depends on the quality of the data gathered and the effectiveness of the analysis and interpretation by the decisionmaker.

One example of the application of a data-driven DST is to use it to support winter road maintenance. Deciding how many crews to send out for plowing and anti-icing operations, to which locations, and at what time of day can be critical functions for areas with winter weather. Using real-time, site-specific reports of pavement and weather conditions and predictions of how road conditions will be affected by forecasted changes in the weather, the data-driven DST in the Kansas City SCOUT TMS uses data to assist maintenance crews within Missouri to schedule and prioritize de-icing and snow removal. This project was started in 2015 to integrate a tool that incorporates real-time and/or archived data with results from an ensemble of forecast and probabilistic models. It fuses them together to predict current and future overall road and travel conditions for travelers, transportation operators, and maintenance providers. This system provides weather advisories to the operators at SCOUT, as shown in figure 17.



© 2015 Kansas City SCOUT.
I = interstate; Rd. = road; W = west.

Figure 17. Screenshot. Notification of potential events by the Kansas City SCOUT.

This project currently provides a stand-alone system, in which the operators within the TMS can view the data and use it to assist them in identifying weather events based on these data. Eventually, this DST will be integrated into the TMS to become a fully integrated and operational subsystem. This system could then be integrated with local road weather information system data for the completion of a DST to support road weather maintenance activities.

CASE STUDY: MODEL-DRIVEN DECISION-SUPPORT TOOL—SAN DIEGO INTEGRATED CORRIDOR MANAGEMENT SYSTEM

As discussed in chapter 2, model-driven DSTs use mathematical models that express theoretical relationships among data elements or key variables of interest during analysis. These tools can be used (either online or offline) to simulate the behavior of a transportation system (or parts of the system) using different values for certain parameters.

The San Diego Interstate 15 DST for the ICM project focused on five primary goals, as follows:

- The corridor’s multimodal and smart-growth approach will improve accessibility to travel options and attain an enhanced level of mobility for corridor travelers.
- The corridor’s safety record will be enhanced through an integrated multimodal approach.
- The corridor’s travelers will have the informational tools needed to make smart travel choices within the corridor.
- The corridor’s institutional partners will employ an integrated approach through a corridorwide perspective to resolve problems.

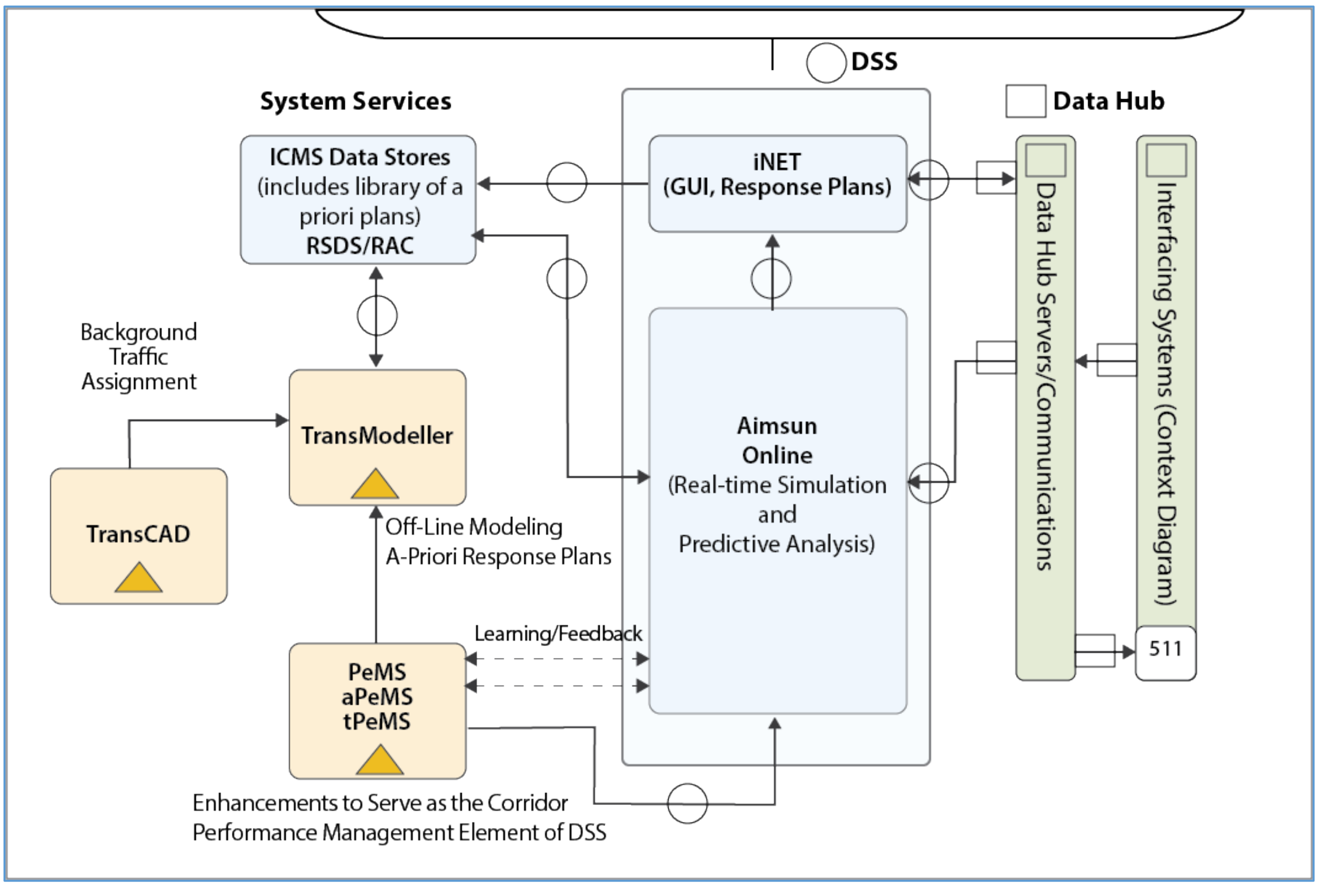
The corridor’s networks will be managed holistically under both normal operating and incident/event conditions in a collaborative and coordinated way.

To achieve these goals, San Diego leveraged existing investments in ITS to implement a TMS that combined road sensors, transit management strategies, video, and traveler information to reduce congestion. The TMS delivered information to commuters via the Internet and DMSs and enabled operations managers to adjust traffic signals and ramp meters to direct travelers to high-occupancy vehicle and high-occupancy toll lanes, bus rapid transit, and other options.

Functionally, the DST component supports the ability to automatically, semiautomatically, or manually generate suggested plans for actions within each mode and agency in response to regional events. The significance and importance of the DST component lie in the fact that the response plans for short- or long-term impacts on the corridor are coordinated and not carried out in isolation, as is usually the case. The DST component relies on a knowledge-driven DST containing information on roadway geometry and field device locations to automatically generate response plans for further evaluation; the TMS then uses a model-driven DST to evaluate the various response plans using a simulation of the network. The simulation model predicts how the corridor and its surrounding areas will perform based on predictions 15 and 30 min into the future. Using a person-throughput through the corridor performance metric, the

DST provides the operator with each option that was simulated along with a performance metric for the operator to select one of the options.

Figure 18 provides a decision-support-centric view of the TMS. At a high level, the TMS includes the DST functionality for three key elements. On the left side of the figure, the orange boxes and the ICM system data store illustrate the offline modeling capability and the response plan database. On the right side of the figure, the green boxes represent the current conditions gathered through the TMS data subsystem, including the APIs and databases. In the middle of the figure, the blue boxes represent the model-driven DST evaluation of response plans through a real-time simulation and predictive analysis. The learning and feedback connection reflects the offline analysis of the real-time simulation analysis using performance measurement system data from the University of California-Berkeley, California Partners for Advanced Transportation Technology, and the California DOT.



iNET = intelligent network; PeMS = performance measurement system; RAC = road asset configuration; RSDS = response strategy data store.

Figure 18. Diagram. Decision-support view of the San Diego integrated corridor management system.

Once an event is triggered in the system (e.g., the congestion threshold has been reached or will be reached soon), the DSS evaluates various response plans. Each response plan consists of predetermined and staged action plans. Once a response plan is selected, the various action plans go into effect at specific times. It is important to note that "event" in the case of the DSS implies a congestion threshold that triggers the evaluation of response plans—the event could be the result of a crash, higher-than-expected volumes, weather, special events, and so on. The reason for the event can be any number of causes (e.g., incidents, work zones, and special events).

Each response plan is also a function of the demand on the corridor and the impact of the event, as shown in table 12, and is defined by the alignments of response postures across each mode based on corridor conditions. A simple explanation of table 12 is that an aggressive response plan is required for a high-impact event during a high-demand condition, whereas a low-impact event during a low-demand situation might require only a conservative response plan.

Table 12. Response plan alignments across demand and event impact.

Type of Demand	Low Impact	Medium Impact	High Impact
Light Demand	Conservative	Conservative	(Event 2) Moderate
Moderate Demand	Conservative	Moderate	Aggressive
Heavy Demand	Moderate	Aggressive	(Event 1) Aggressive

REFERENCES

Alexiadis, V., and K. Jeannotte and A. Chandra. 2004. *Traffic Analysis Toolbox Volume I: Traffic Analysis Tools Primer*. Report No. FHWA-HRT-04-038. Washington, DC: FHWA.

Altschuld, J. 2010. *Needs Assessment: An Overview*. Thousand Oaks, CA: Sage Publications, Inc.

Cyert, R.M., and J.G. March. 1963. *A Behavioral Theory of the Firm*. Englewood Cliffs, NJ: Prentice-Hall.

Dowling, R. 2007. *Traffic Analysis Toolbox Volume VI: Definition, Interpretation, and Calculation of Traffic Analysis Tools Measures of Effectiveness*. Report No. FHWA-HOP-08-054. Washington, DC: FHWA.

FHWA. 2005. *Transportation Management System Performance Monitoring, Evaluation, and Reporting—A Technical Handbook*. FHWA-HOP-07-124. Washington, DC: FHWA.

FHWA and California Department of Transportation. 2009. *Systems Engineering Guidebook for Intelligent Transportation Systems 3.0*. Washington, DC: FHWA. https://www.fhwa.dot.gov/cadiv/segb/files/segbversion3.pdf, last accessed April 23, 2019.

FDOT. 2005. *Statewide Systems Engineering Management Plan for Intelligent Transportation Systems*. https://www.fdot.gov/docs/default-source/traffic/its/projects_deploy/semp/pdf/050315_D1-10_V2.pdf, last accessed February 2020.

INCOSE. 2012. *INCOSE Systems Engineering Handbook*, version 3.2.2. INCOSE-TP-2003-002-03.2.2. San Diego, CA: INCOSE.

IEEE. 1998. *IEEE 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. New York, NY: IEEE.

IEEE. 2006. *IEEE 1512 Standard For Common Incident Management Message Sets For Use By Emergency Management Centers*. New York, NY: IEEE.

IEE Standard Association. 2011. *ISO/ International Engineering Consortium/IEEE 29148:2011 Guidance for the Development of the Set of Requirements, System Requirements Specification (SyRS)*.

Institute of Transportation Engineers. 2020. *Traffic Management Data Dictionary (TMDD) Standard v3.1 for the Center to Center Communications*. Washington, DC: Institute of Transportation Engineers.

Margiotta, R., T. Lomax, M. Hallenbeck, S. Turner, A. Skabardonis, C. Ferrell, and B. Eisele. 2007. *Guide to Effective Freeway Performance Measurement: Final Report and Guidebook*. Washington, DC: The National Academies Press.

Myers, A. 2018. "The Australian Approach and Architecture for Decision Support Subsystems." Presented at the *TRB Annual Meeting.* Washington, DC: TRB.

National Academies of Sciences, Engineering, and Medicine. 2016. *Highway Capacity Manual*, 6th ed. Washington, DC: TRB.

Robinson, E., and M. Motamed, D. Newton, K. Olyai, and L. Bedsole. 2017. *Elements of Business Rules and Decision Support Systems Within Integrated Corridor Management: Understanding the Intersection of These Three Components.* Report No. FHWA-HOP-17-027. Washington, DC: FHWA.

Simon, H.A. 1955. "A Behavioral Model of Rational Choice." *The Quarterly Journal of Economics* 69, no. 1: 99–118.

Tavasszy, L.A. 1998. "A DSS for Modelling Logistic Chains in Freight Transportation." *International Transactions in Operational Research* 5, no. 6: 447–459.

TRB. 2013. *Evaluating Alternative Operations Strategies To Improve Travel Time Reliability*. Strategic Highway Research Program (SHRP 2) Report S2-L11-RR-1. Washington, DC: National Academies Press.

UK Office of Government Commerce. nd. "ITIL Open Guide" (website). https://www.itlibrary.org/index.php?page=CCTA, last accessed September 17, 2021.

HRDO-20/11-21(WEB)E