Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

MANAGING COMPLEX DATA FLOWS IN THE WISCONSIN LANE CLOSURE SYSTEM

Michael J. Runnels
Systems Developer, TOPS Lab
UW-Madison
2205 Engineering Hall
1415 Engineering Drive
Madison, WI 53706
(608) 265-2366
mrunnels@engr.wisc.edu

Steven T. Parker, Ph.D.
IT Program Manager, TOPS Lab
UW-Madison
2205 Engineering Hall
1415 Engineering Drive
Madison, WI 53706
(608) 265-4921
sparker@engr.wisc.edu

David A. Noyce, Ph.D.
Director, TOPS Lab
UW-Madison
2205 Engineering Hall
1415 Engineering Drive
Madison, WI 53706
(608) 265-1882
noyce@engr.wisc.edu

Yang Cheng
Research Intern, TOPS Lab
UW-Madison
2205 Engineering Hall
1415 Engineering Drive
Madison, WI 53706
cheng8@wisc.edu

Traffic Operations and Safety (TOPS) Laboratory
Department of Civil and Environmental Engineering
University of Wisconsin – Madison
1415 Engineering Drive
Madison, WI 53706

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

ABSTRACT

The Wisconsin Lane Closure System (WisLCS) serves as the central acceptance and reporting system for Wisconsin DOT (WisDOT) lane closures and restrictions statewide. It was developed by the Traffic Operations and Safety (TOPS) Laboratory at the University of Wisconsin-Madison through sponsorship by WisDOT Bureau of Highway Operations to address several emerging needs with respect to work zone operations and data reporting. In particular, the WisLCS will provide lane closure information to the Wisconsin 511 Traveler Information system when it launches in 2009. The WisLCS system was developed over 18 months and has been operational since April 1, 2008.

This paper describes how the Wisconsin Lane Closure System (WisLCS) utilizes a data-centric approach to enhance the functionality of the lane closure process and open it up to related systems. The WisLCS was designed to enhance closure operations at WisDOT by tracking a wider range of closure types compared to existing systems and by providing more dynamic, real-time closure status information. Our challenge was to take the many different data needs arising from these requirements and build a system robust enough to handle a complex array of data elements at the same time developing user interfaces that both collect and present the data in a straightforward and logical fashion. The approach taken by WisLCS was in the development of a lane closure data model, implemented as a collection of closure data objects. These data objects encapsulate all the data elements necessary to capture every aspect of a closure, including traditional "static" elements such as time, location, and project details, and more dynamic elements such as status, history, and authorization attributes.

An important feature of the WisLCS closure data model is its integration with the Wisconsin Department of Transportation (WisDOT) State Trunk Network (STN) GIS. Developing WisLCS to the STN allows our system to 'push' lane closure information to other systems in a standardized and easily integrated format. It also facilitates integration with internal transportation applications at the Traffic Operations and Safety (TOPS) Lab. For example, the WisLCS Capacity module utilizes STN-linked data from an established WisDOT vehicle volume data source and helps to determine appropriate thresholds for use in the lane closure decision making process. Moreover, the STN based data model is intended to facilitate future development of map-based reporting and interface tools.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

Wisconsin Lane Closure System Overview

The Wisconsin Lane Closure System (WisLCS) serves as the central acceptance and reporting system for Wisconsin DOT (WisDOT) lane closures and restrictions statewide. It was developed by the Traffic Operations and Safety (TOPS) Laboratory at the University of Wisconsin-Madison through sponsorship by WisDOT Bureau of Highway Operations to address several emerging needs with respect to work zone operations and data reporting. In particular, the WisLCS will provide lane closure information to the Wisconsin 511 Traveler Information system when it launches in 2009. The WisLCS system was developed over 18 months and has been operational since April 1, 2008.

This paper covers technical design aspects of the WisLCS as an information system, in particular with regard to the WisLCS data model. Indeed, the essential design philosophy that governed the development of the WisLCS can be characterized as a *data-centric* one. That is, the various data requirements – in terms of physical elements and interactions between data – were described first. Following the creation of the data model, the WisLCS interfaces and capabilities were developed in order to support requirements of the data model.

The WisLCS is a super thin client web-based application. Users interface with the WisLCS through a web browser and web-page based forms and displays. Most of the processing (business logic) is performed on a centralized server at TOPS lab. As such, there are minimal system requirements on the part of the end-user. In fact, some users interact with WisLCS over dial-up modem from workzone field offices. It is also a user-authenticated system that is structured so that individuals have differing access levels based on a role and region hierarchy. The main purpose of the WisLCS is to handle upcoming lane closure operations and provide real-time adjustment and reporting of those closures. The 'actions' a user can take on any given closure depends upon different closure and/or user attributes. Here is a breakdown of the different schedule actions a user may take.

1. (Accept Interface):
   a. ACCEPTED: A closure is now 'live' and reportable within the system.
   b. RETURNED: A closure that is returned has not been edited but requires modification before acceptance.
   c. RESCHEDULED (PENDING): An accepted closure has been returned to the Accept Interface for rescheduling.
   d. DELETED: A closure is no longer valid or viewable in the system.
   e. CANCELED: A closure or one of its facilities is moved to a final status of canceled and no further actions can take place for them.
   f. PENDING: A closure has been forwarded to a specific role in the system for review prior to acceptance.
   g. ENTERED: A new closure that is initially entered into the system.
2. (Modify Interface):
   a. RESCHEDULE: An accepted closure is about to be returned to the Accept Interface for rescheduling.
   b. MODIFY: Any closure can be modified (Modify General, Modify, Modify Date/Time) by a user with acceptance authorization. The closure status remains accepted.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

      c. MODIFY PENDING ACCEPTANCE: This is a special case modify for any closure that is modified by a user without acceptance authorization. Only certain 'pending' date/time fields are entered.

      d. CANCELED: A closure or one of its facilities is moved to a final status of canceled and no further actions can take place for them.

      e. COMPLETED: A closure or one of its facilities is moved to a final status of completed and no further actions can take place for them.

        Closures have a life cycle that is flexible in that it can be extremely concise (entered with acceptance then left to run its course) or complex; and go through any number of the actions mentioned. It is an interface-based system where each interface serves as a logical grouping of actions to view, report, compare, or move closures through their life cycle. In relation to the amount of data handled in a 'real time' and dynamic manner (both inputs and outputs), it is an extremely fast and secure system. Since coming online April 1st, 2008 the WisLCS has added approximately 625 new closures a month and as of this papers submission date there are 621 users.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

FIGURE 1 Reports Interface – Users can quickly view and modify active closures.

Figure 1 above illustrates the WisLCS Reports Interface. At the very top is common system header informational links and navigation links (to the various interfaces). The Reports Interface itself consists of 3 sections; the Search section, the Search/Results Information section, and then the Results section. Within the Search section a user may choose from any number of system-allocated and/or generated selections which are then used to query the database for the

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

proper closure data. In turn this data is then formatted and displayed in the Results section. On the surface many of these selections may seem straightforward and innocuous however a great deal of data-centric logic is being done prior to selections, after certain selections, and upon submission. For example the list of highways in that particular selection dropdown is populated from a list of those available from the locations database. Moreover the highway list dynamically changes depending upon the Project ID and Region/County selections. Searching on a highway hides a lot of the system complexity because every closure maintains data which links it to a highway and all counties it intersects; along with any possible highway combinations.   Many of the selections throughout the system are dynamic in this fashion.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

Taking a Data-Centric Approach in the WisLCS
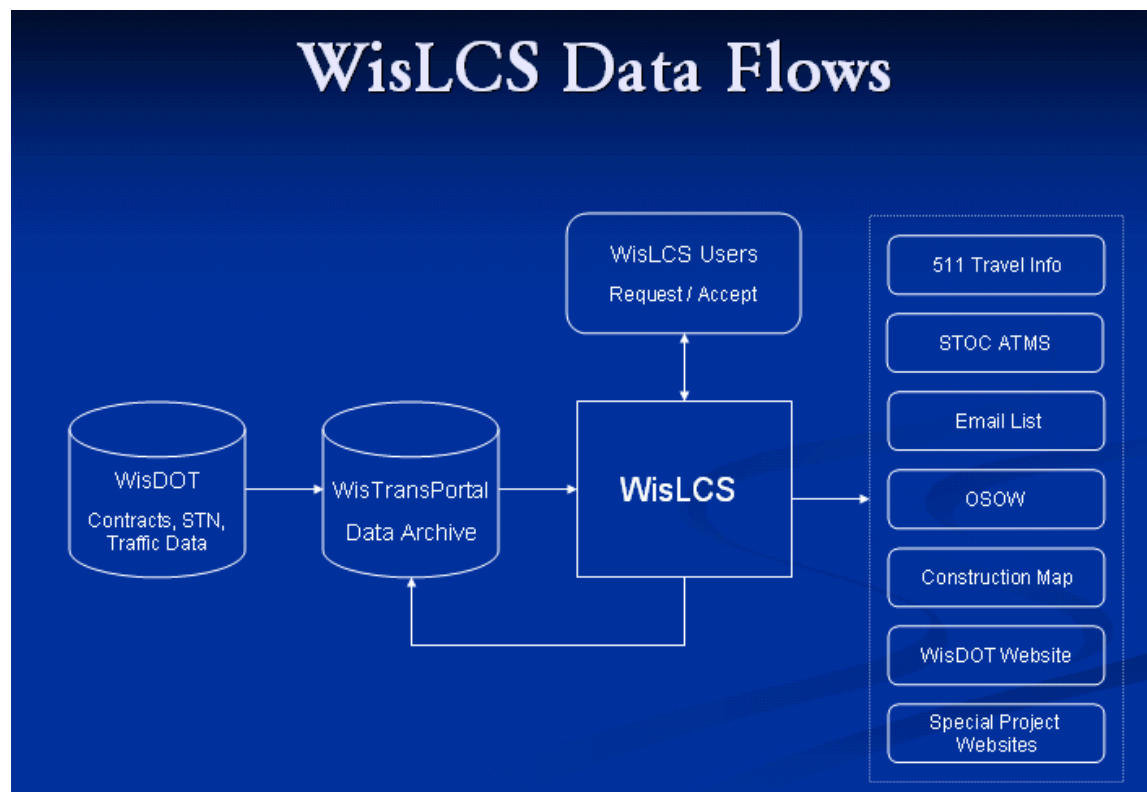
FIGURE 2 Shows the various inputs and outputs of the WisLCS.

Data-Centric is a generic term that we use in reference to our WisLCS system design philosophy and our approach to building its functionality and user interfaces. The lane closure system centers on data; how data is entered, how data is secured, how data is retrieved and displayed, how data interacts with other systems. Many challenges face a system that is so dependent upon data. The risks that are to be avoided or minimized within the WisLCS include:

- Data Entry Errors
- Incomplete Data
- Non-Standard Location Information
- Outdated Information
- Data Performance Issues
- Accountability of Data Entry/Modifications
- Aligning Data With Current Business Practices

As shown in figure 2 above the WisLCS has a number of data inputs and data outputs. Some of the input data needs to be filtered and run through a series of processing steps to refine it before it can be used by the WisLCS. Most of the output data also needs filtering and refining before it can feed other systems. The efficiency and accuracy of all the systems in the data flow depend on the WisLCS providing data that is complete, accurate, standard, up-to-date, and precise. In order to eliminate the risks mentioned above and to achieve the high quality of input/output data flows a series of data-centric practices were implemented. The WisLCS has many components that make up its data model and relate to a closures life cycle. These

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

components can be either traditional static data such as time and location or have a more dynamic attribute such as in the case of a closures current status or its history. Specifically we utilized a data-centric approach to design and build the system around its major components which are:

- General Closure Information
- One-To-Many Facilities
- General and Facility Closure Location(s)
- Closure History/Status
- User Roles and Authorizations

Taking a data-centric approach was a logical choice based on the fact that all of these components are made up of data objects and elements. The data in question is such that it is part of many different data flows. Data flows that feed other systems and interact with other data flows within the system. In opposition to these types of dynamic and interdependent data flows would be static essay type data flows such as one would find in an online survey form. That data is not vetted for accuracy or subject to standards and does not interact with other data. Seeing as how this is not the type of data the WisLCS builds its data flows around, it is not only logical but crucial to implement the data-centric practices to be outlined in this paper.

GENERAL TABLE

NAME	TYPE	PK	DEFINITION
CID	NUMBER	X	Unique Closure ID
CLOSURE_TYPE	CHARACTER		Closure type. One of Construction, Maintenance, Permit, Special Event, or Emergency.
PROJECT_DESC	CHARACTER		General closure description.
PROJECT_ID	CHARACTER		WisDOT Project ID for construction project.
PERMIT_ID	CHARACTER		Utility work permit ID.
WZ_MAP_NO	CHARACTER		Identifier for closure if included on the WisDOT Work Zone map.
OMIT_FROM_MAP	BOOLEAN		Flag indicating that a closure should not be included on the WisDOT Work Zone map.
LOCAL_PRGM	BOOLEAN		Flag indicating a Local Program construction project.
EXP_IMPACT	BOOLEAN		Expected impact of a workzone.
PRIMARY_CONTACT_NAME	CHARACTER		Primary contact information - name
PRIMARY_CONTACT_PHONE	CHARACTER		Primary contact information - phone
SEC_CONTACT_NAME	CHARACTER		Secondary contact information - name
SEC_CONTACT_PHONE	CHARACTER		Secondary contact information - phone
CONTACT_ETCC_NAME	CHARACTER		Emergency traffic control contractor contact information - name
CONTACT_ETCC_PHONE	CHARACTER		Emergency traffic control contractor contact information - phone
CONTACT_LE_NAME	CHARACTER		Law enforcement contact information - name
CONTACT_LE_PHONE	CHARACTER		Law enforcement contact information - phone
CONTACT_OTHER_NAME	CHARACTER		Other contact information - name
CONTACT_OTHER_PHONE	CHARACTER		Other contact information - phone
INTERNAL_COMMENT	CHARACTER		Text field for WisLCS internal comments
EXTERNAL_COMMENT	CHARACTER		Text filed for reportable (external) comments.

FIGURE 3 Excerpt from the WisLCS Data Dictionary.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

At the outset of developing a data-centric system it is necessary to create a data-model with a complete set of data elements (see figure 3). These data elements must encompass all the necessary information used by the system and its outputs. Furthermore, each data element must normalize within the system in order to create data that is both standardized and wholly encompasses the information that is stored within. Figure 3 shows a snapshot of the contents of the WisLCS Data Dictionary. Within that document's contents are the name, description, and attributes for every data element in the system. The way these data elements are grouped and used in the system lays the foundation for the data-centric approach taken in the system design and development.

At the core of the system is this complete set of data elements. These data elements are then grouped into logical areas to form data objects. Data objects are analogous to the system components mentioned above and within the database itself as database tables. However data-centric is more than just database design. There are many types of complex data flows and issues that a data-centric approach attempts to manage. In the lane closure system, data issues are managed through the use of standardized data sets, encapsulating data elements into objects (beans), and the cross-utilization of data elements. These are the data-centric approaches talked about in this paper and that will not only enhance the functionality of the WisLCS but also help to share information with other outputs or systems.

As figure 4 below shows; the varying user roles in the system hierarchy enables which interfaces a role has access to and moreover within those interfaces which actions a user is allowed to enact. The combination of roles and actions in the various interfaces represent many differing data flows within the system. With these differences comes a great challenge to streamline the system data into a cohesive and easily understood grouping of data objects.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

Interface/Role/Actions Matrix

Interfaces/Roles

	Home	Request	*Accept	*Modify	*Reports	Search	Capacity	*Calendar	Email	Preferences	*Admin
Supervisor	x	x	x	x	x	x	x	x	x	x	x
RTE	x	x	x	x	x	x	x	x	x	x	x
STOC	x	x	x	x	x	x	x	x	x	x	x
PM	x	x	x	x	x	x	x	x	x	x	x
MC	x	x	x	x	x	x	x	x	x	x	
PC	x	x	x	x	x	x	x	x	x	x	
PL	x	x	x	x	x	x	x	x	x	x	
Inspector	x	x	x	x	x	x	x	x	x	x	
Public Information	x				x	x	x	x	x	x	x*
Requestor	x	x	x	x	x	x	x	x	x	x	
Viewer	x				x	x	x	x	x	x	
Sys Admin	x				x	x	x	x	x	x	x

*Accept – The actions available within the Accept Interface are based upon role and having acceptance authority
*Modify – The actions available within the Modify Interface are based upon role and having acceptance authority
*Reports – The actions available within the Reports Interface are based upon role and having acceptance authority
*Calendar (Events) – Supervisor, RTE, STOC, and Public Information role will have actions available to add/update/delete Events
*Admin – Public Information has access to Email in the Admin

Accept Interface – Actions/Roles

	Edit	Return	FT Sprv	FT RTE	FT STOC	FT PM	FT MC	FT PC	FT PL	FT Insp	Accept	Delete	Cancel
Supervisor	x	x									x (w/aa)	x	x
RTE	x	x	x		x						x (w/aa)	x	x
STOC	x	x	x	x							x (w/aa)	x	x
PM	x	x	x	x	x						x (w/aa)	x	x
MC	x	x		x	x						x (w/aa)	x	x
PC	x	x		x	x						x (w/aa)	x	x
PL	x	x		x	x	x					x (w/aa)	x	x
Inspector	x	x		x	x	x	x	x	x		x (w/aa)	x	x
Requestor	x			x		x	x	x	x	x	x (w/aa)	x	x

*(w/aa) – with acceptance authority

FIGURE 4 The Interface Roles Matrix shows the challenges to making a universal system conform to data standards.

Though the above figure 4 is representative to showing many of the possible data flows within the system and with each data flow comes the challenges and risks mentioned earlier to building a dynamic yet manageable system; it is easiest to understand the data-centric approach by describing our chosen solutions.

Cross-Utilization of data is an underpinning objective of the WisLCS. The WisLCS was designed with the understanding that it would utilize 'outside' data from multiple data sources and that in turn it would be feeding other outputs and systems such as *511 Traveler Information, Email Distributions, and XML for Websites.* Moreover the data in the WisLCS should be made interactive within the system. Closures can be cross-matched against other closures and special events in order to find possible conflicts or to coordinate closures of different types; the benefits being shorter closure times as different groups 'piggyback' their closures and also limiting the amount of closures allowed on a heavily traveled network. Another useful cross-utilization of data through the use of the WisLCS Capacity Module to look up volume thresholds, *Monthly Average Volume (per day of week, per hour)* data that in turn will help users determine available closure time windows (shown in figure 5 below).
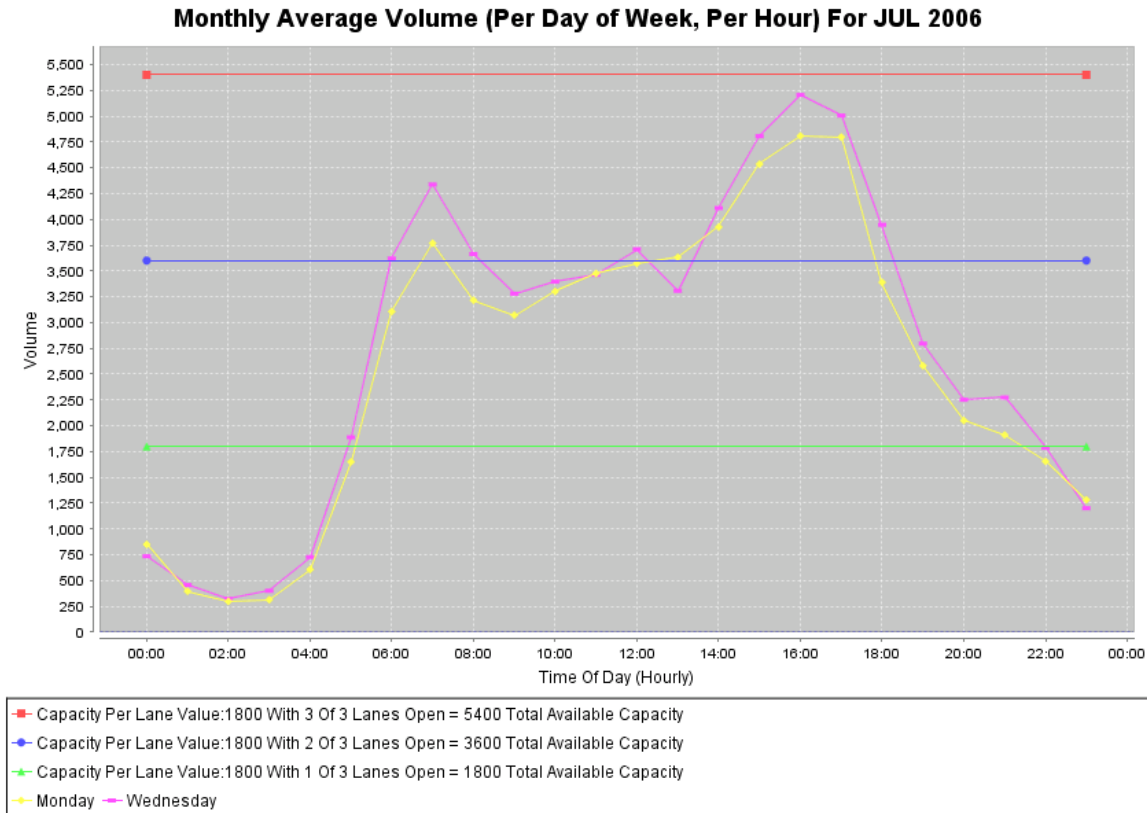
Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

Monthly Average Volume (Per Day of Week, Per Hour) For JUL 2006

Legend:
- ■ Capacity Per Lane Value:1800 With 3 Of 3 Lanes Open = 5400 Total Available Capacity
- ● Capacity Per Lane Value:1800 With 2 Of 3 Lanes Open = 3600 Total Available Capacity
- ▲ Capacity Per Lane Value:1800 With 1 Of 3 Lanes Open = 1800 Total Available Capacity
- Monday — Wednesday

FIGURE 5 Capacity Volume Threshold Graph.

Closures can be scheduled during lower traffic hours, providing better throughput and safer workzones. With all the raw data available from both internal and external inputs; the cross-utilization strategies employed turn these complex data flows into an opportunity for data management and productivity.   The cross-utilization in these varying ways helps to manage the complex flow of data from many sources and multiple closures.

Another of the data-centric approaches taken was to create a complete set of data elements in a unified manner. The challenge was to achieve this task without the unwieldiness of overloaded user input interfaces that differed significantly amongst closure types and allow a user to group closures. This was accomplished through the strategies and methods below:

1. *Encapsulation*. By encapsulating data elements into logical data objects the user experiences another level of system simplicity. Closures are data objects that consist of other data objects; the General section object, one-to-many Facility section objects, and multiple Facility History objects. The grouping of data elements in this manner serves a number of purposes. First it makes storage and retrieval of closures quick and efficient. The one-to-many grouping of facilities will 'shield' the user from the unnecessary duplication a one-to-one general/facility closure would require. And it organizes data into logical areas. Let's take for example a simple closure like the one pictured below (collapsed view) in figure 6.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

FIGURE 6 Collapsed View of a Typical Construction Closure.

This helps to illustrate that a typical closure as an encapsulated object is made up of a General section object, one-to-many Facility section objects, and multiple Facility History objects. By grouping the data elements in this fashion we are able to translate the closure data elements into database table 'elements' or columns. Each of the objects stated is represented by a database table and each specific closure object is one or more easily identified rows in those tables. This makes storage and retrieval of this information extremely fast. It also allows closures of all types to be easily filtered or searched for by virtually any of the closure elements or combination thereof. As an example we could search for and filter only those closures that take place in the county of DANE along US 12 EB and whose current status is scheduled but not accepted (so entered, pending, or returned). Because the data elements are 'grouped' at the database level no data mining or conversion needs to happen and because the data elements are standardized the search will find closures of all types (Construction, Maintenance, Permit, etc.). The gross number of data elements associated with closures make-up a complex and unorganized jumble of data. The encapsulation techniques used create simplified and organized flows of that data.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

In figure 7 below the field values shown are the dynamically set values derived from both system defaults and user interaction. Although not shown, the other main closure types (Maintenance and Permit) have the same look and feel. The interface fields are identical to Construction except when it is absolutely necessary (because of differing business requirements); then system functionality is used to dynamically change the interface ever so slightly to meet those business needs while maintaining the overall standardized look and feel which reduces the confusion of multiple non-similar interfaces and prevents data entry error of non-relevant fields.

Closure Request

CONSTRUCTION REQUEST - Asterisk(*) indicates *required* field.

*Project ID	*Begin County	*Primary Contact:	*Phone:
1000-19-63	WAUKESHA	Cruz Bernal-Albano (cbernal/ALL)	
*General Description	*End County	Secondary Contact:	Phone:
SIGN BRIDGE REPAIRS 2008 | VARIOUS FREEWAYS, STATE HIGHWAYS AND US HIGHWAYS |	RACINE	Judy Geurts (DOTJLG/ALL)	
	*Hwy	* Emergency Traffic Control Contractor Name:	* Phone:
	I-94 EB		
Local Program	*Duration [?]	Law Enforcement Name:	Phone:
☐	Daily/Nightly		
	Expected Impact	Other Contact Name:	Phone:
	LOW		

Internal Comment	External Comment	Near Region Boundary [?]
		SW SE NE NC NW
		☐ ☐ ☐ ☐ ☐
		(do not check origin region)

FIGURE 7 Construction Closure Request showing the standardized and dynamic nature of the General section.

2. *Standardization*. Not all closure types have the same data requirements. We formed data elements that could encapsulate these differing requirements. We compromised on the verbiage and format of the elements in order to achieve standard data elements capable of unifying the closure types into using a single closure object. As an example, the Construction and Maintenance closure interfaces both have an Expected Impact field. For Construction projects this was thought to mean the amount of delay the closure would cause a traveler. For Maintenance it was more of a traveler inconvenience value. The compromise was to find a set of standard values and the nomenclature that both types could understand. In addition, the set of values are static, allowing the user to pick from a dropdown so that it is easier and not subject to text entry issues. The Expected Impact field is now unified so that when a Construction type user is looking over a Maintenance closure they can understand the Expected Impact that closure will cause. Though the system has to handle closures of different types, it incorporates first encapsulation of all the various data elements and standardizes them to manage the internal flow of data between these different closures.

The inherently complex standardizing part comes into play for those requirements that are incompatible with other closure types. For example a Construction type has need of a Project ID (see figure 7 above) whereas the other types cannot use this field. This is where the next method comes into play.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

3. *Dynamic User Interface*. Creating dynamic data elements comes in a few flavors. For the case of Project ID and Permit Number these are both members of the closure object AND are required fields on the Request interface. However, the Closure Type field triggers a dynamic Request interface that is 'built' with or without the Project ID or Permit Number fields depending on the Closure Type value. The advantage to this is that a user entering a Construction closure request doesn't have to worry about filling out or ignoring a Permit Number field since it doesn't show up on the interface in the first place. Another dynamic feature within the system is based on data element interdependencies. Some fields within the system are dropdown lists whose values are derived from the selections of other fields. For example the Hwy field dropdown list is based on the selections within the Begin County and End County fields. Only those highways that intersect with both county selections will be shown in the Hwy list. Other dynamic features within the system include enabling/disabling fields and making fields required based on data selections. Integrating the data elements in this dynamic fashion provides the following benefits to the system users:
   a. Allows users to use the dynamic nature of the system as another source of information. For instance if a user wanted to see a list of all the highways that intersect with Green county they could use the county dropdown to activate the dynamic highway dropdown list
   b. Makes input selection simpler by narrowing options and eliminating user error
   c. Many input fields are pre-filled with default values from dynamic actions. This translates to simpler more intuitive user interfaces
   d. Provides more real-time selections. Many of the dynamic actions use the most recent values from our distributed database

Through Encapsulation, Standardization, and Dynamic User Interfaces the system overcomes the complex flow issues of data that is unorganized, ungrouped, has limited use, cannot be easily searched or filtered, and grows rapidly to an unmanageable size.

Finally, turning the focus more closely on the relationship between the WisLCS and the STN (State Trunk Network) is a primary example of how to leverage data-centric policies to best capitalize on outside data sources. Integrating the STN into the WisLCS posed many challenges but also creates the following benefits:

The STN is a WisDOT standard dataset used in many existing systems; by using the STN we are able to create input/output data flows that conform to existing standards. Because the STN is used throughout WisDOT, users have a familiarity with the elements and information that make up the STN. Also the STN is maintained outside the WisLCS so that regular updates can be coordinated. The STN has recently been expanded to incorporate longitude and latitude, setting the stage for GIS mapping features in the future.

Michael J. Runnels   Steven T. Parker   David A. Noyce   Yang Cheng

Conclusion

There are many inherent risks associated with a system of this complexity and one that contains various and numerous data flows that interact and depend on one another. These risks include: possible data entry errors, incomplete data, non-standard information, outdated information, data performance issues, accountability of data entry or data modifications, and aligning data with current business practices. Within a system such as the WisLCS that relies so heavily upon performance and accuracy it is imperative to eliminate or mitigate these issues. As shown, the use of several data-centric approaches and practices can overcome these obstacles. By implementing encapsulation, standardization, and providing dynamic user interfaces the WisLCS becomes a fast, accurate, and secure system used throughout the state of Wisconsin and moreover will feed current and future outputs in kind.

References

1. Wisconsin DOT, TOPS Laboratory UW-Madison. Lane Closure System.
   http://transportal.cee.wisc.edu/closures/
   Accessed July 31, 2008