Automated Safety Warning Controller (ASWC) Phase I – Proof of Concept

Doug Galarus, Dan Richter and Kelvin Bateman
Western Transportation Institute – Montana State University

Sean Campbell
Caltrans Division of Research and Innovation

Ian Turnbull
Chief, Office of ITS Engineering and Support, Caltrans District 2
Caltrans posed the following problem statement:

*In order to provide better safety warning information to motorists, how can roadside condition sensor data be automatically analyzed and real-time road condition information be displayed to the traveling public?*

[Caltrans Research Problem Statement]
Automated warning systems are not a new concept within the transportation community. There are several projects on the state highway that use the concept of a roadway sensor initiating some type of motorist warning. To date, all of these systems are unique implementations that use one-of-a-kind software for control. The system controller is a custom device which can only be used with that particular project's physical and electrical layout. The department has benefited from a standardized approach to individual field elements such as CMS, EMS and detection loops. A standardized automated warning system controller, which controls standardized field elements in a system environment, has not been developed to date.

[Caltrans Research Problem Statement]
Current Practice

Manual data retrieval and analysis at TMC
- Someone in the field notifies Dispatch of inclement conditions
- Dispatch notifies the TMC
- The TMC operator verifies conditions on RWIS
- The TMC operator manually sets the warning (CMS, EMS, Flashing beacon)
- The warning stays in place until the TMC operator is informed of a change or he notices it himself

Alternatively:
- The TMC operator is expecting inclement weather and watches the current conditions
- Puts the warning up when conditions get bad enough
- The warning stays in place until the TMC operator is informed of a change or he notices it himself

Problems:
- Manual process: requires people
- Latency: delay between conditions worsening and warning being put up
- Long connection distance: higher chance of failure
- Human error
- Off hours issues – TMC not manned 24/7
- Process often breaks down at the warning removal step
Current Practice

• One of a Kind Controllers
  – Purpose built for a single type of warning
  – A complete package (sensors, signs, and other hardware)

• Problems:
  – Only one type of warning
  – Requires installation of single-use hardware instead of re-using existing sensors, signs, etc.
  – Non-standard: hardware, interface may vary from site to site
Automated Safety Warning Controller
Phase I Concept

- General purpose controller
- Uses existing sensors
  - RWIS, Loop Detector, MVDS
- Uses existing displays
  - CMS, EMS, Flashing Beacon
- User configurable by Field Engineers
  - Flexibility to specify devices for individual site configuration
  - Users can write Alert Scripts to meet the needs of the site

- Advantages
  - Located at site, short network length
  - Frequent poll and evaluation of data
  - 24/7/365
  - Standard for all sites and all uses
Basic Concept Summary

Diagram showing TMC connected to Controller, which has Data Collection Field Elements (Loop Detector, RWIS, MVDS) and Information Output Field Elements (CMS, Flashing Beacon, EMS).
System Data Flow

1. RWIS records data
2. Module collects data
3. Module stores data
4. Script retrieves and analyzes data, determines alert is in effect
5. Message for a CMS is stored
6. Module checks for messages in storage
7. Module puts message on CMS
Field Elements

- Road Weather Information Systems (RWIS)
- Changeable Message Signs (CMS)
- Extinguishable Message Signs (EMS)
- Highway Advisory Radios (HAR)
- Loop Detectors
- Microwave Vehicle Detection Systems (MVDS)

- Protocols
  - RWIS: SNMP with NTCIP defined OIDs
  - CMS: proprietary byte stream
  - EMS, Flashing Beacon: HTTP (WebRelay)
  - Loop Detector: Proprietary byte stream
Platform

Operating System:
Embedded Linux - Montavista

Programming / Scripting Language:
Python

MOXA UC-7420

MOXA DA-661-16-LX
Testing Lab Setup

Thanks to Caltrans and Ian Turnbull
Simulate Caltrans field elements and communication via elements and TMC.

Equipment includes:
- Two CMS Controllers
- Loop Detector
- SNI Servers
- Power Supplies
- Modems and Routers
- NTCIP exerciser software to simulate RWIS

WTI Systems Lab
- Phone lines for dial up networking
- Communication tower on roof
- Private network separate from MSU/public network
  - Connectivity to Systems Staff Offices, Labs and Rooftop

Caltans Loaner Equipment
Communications equipment
Loop Detector (Model 222 GP5 in a DTS 170e Controller)
CMS (SignView 170, Model 500)
CMS Module Logic

1. Wait for run interval
2. Check/delete expired messages in message pool from alert scripts
3. Select highest priority message from message pool
4. Get last message we sent to sign from CMS history file
5. Last message not ‘blank’ and new message from pool?
   - yes: No new message and the last message we sent was ‘blank’ so do nothing
   - no:
5.1 Save deleted message indicating it was not displayed
CMS Module Logic

Can we communicate with sign?
  yes
  Get message from sign
  no

Message on sign is ours or ‘blank’?
  yes
  Message on sign is expired or blank?
  no
  TMC message is on sign; log to system log and CMS log
  no

New message is higher priority?
  yes

Send new message or ‘blank’ (if current message has expired) to sign and CMS log. Save copy of message in CMS data file
  no

Log error to system log
windwarning:
{
    status : 1
    initial_delay : 120
    interval : 60
    timeout : 2
    threshold:
    {
        highwindwarning : 56
        windwarning : 41
        windgustwarning : 46
        windadvisory : 26
        WindDistanceWest:5
    }
}

icewarning:
{
    status : 1
    initial_delay : 120
    interval : 60
    timeout : 4
    threshold:
    {
        freezetemp : 32.0
        precipforice : 1.25
    }
}
Administration

• Four levels of security:
  – Operator
    • Monitor behavior and check for errors, but no changes permitted
  – Supervisor
    • Edit threshold variables
    • start/stop/pause modules
  – Technician
    • Modify alert scripts
    • Edit configuration files
    • Add/remove field elements
  – Administrator
    • Root access, full control over the device
Take a drive through a potential application site …

Photos from Google Street View
Take a drive through a potential application site ... Photos from Google Street View
Pilot Test - Spring Garden

Photos from Ian Turnbull
Pilot Test - Spring Garden
Pilot Test - Spring Garden

Photos from Ian Turnbull
Pilot Test - Spring Garden

Surface Sensors
Photo from Doug Galarus

RWIS
Photo from Ian Turnbull
Pilot Test - Spring Garden

Surface Sensor

Photo from Dan Richter
Pilot Test - Spring Garden

Automated Controller

Photos from Doug Galarus
Ice Warning Script for Spring Garden

#Name: Icy Curve Alert Script
#Reference: Caltrans District 2 Icy Curve Alert Parameters
#Date: 12-04-2008
#updated 25Aug2009 K. Beals
# added additional pucks and second CMS control
# added ice watch surface condition
# updated 11Dec2009

# changed sign message to "CAUTION ICY ROAD" in double stroke, flashing
# Gerry Reyes determined message

ICEWARN = 7               #NTCIP essSurfaceStatus value for ice warning
ICEWATCH = 8              #NTCIP essSurfaceStatus value for ice watch
FROST = 13                #NTCIP essSurfaceStatus value for frost

SurfaceIceWarning = False
SurfaceIceWatch = False
SurfaceFrost = False

SignTest = test           # value can be changed from TMC for testing CMS displays

SurfaceIceWarning = RWIS.essSurfaceStatus1()==ICEWARN or \    
                   RWIS.essSurfaceStatus2()==ICEWARN or \    
                   RWIS.essSurfaceStatus3()==ICEWARN or \    
                   RWIS.essSurfaceStatus4()==ICEWARN or \    
                   RWIS.essSurfaceStatus5()==ICEWARN or \    
                   RWIS.essSurfaceStatus6()==ICEWARN
Ice Warning Script for Spring Garden

SurfaceIceWatch = RWIS.essSurfaceStatus1()==ICEWATCH or \
    RWIS.essSurfaceStatus2()==ICEWATCH or \
    RWIS.essSurfaceStatus3()==ICEWATCH or \
    RWIS.essSurfaceStatus4()==ICEWATCH or \
    RWIS.essSurfaceStatus5()==ICEWATCH or \
    RWIS.essSurfaceStatus6()==ICEWATCH

SurfaceFrost = RWIS.essSurfaceStatus1()==FROST or \
    RWIS.essSurfaceStatus2()==FROST or \
    RWIS.essSurfaceStatus3()==FROST or \
    RWIS.essSurfaceStatus4()==FROST or \
    RWIS.essSurfaceStatus5()==FROST or \
    RWIS.essSurfaceStatus6()==FROST

CMSEast.MessageType = Page1Flash
CMSEast.FontPage1 = DoubleStroke
CMSEast.DisplayTime = 10     #in 1/10 seconds
CMSEast.Priority = 10
CMSEast.Expiration = 20 * MinuteInSecs

CMSWest.MessageType = Page1Flash
CMSWest.FontPage1 = DoubleStroke
CMSWest.DisplayTime = 10     #in 1/10 seconds
CMSWest.Priority = 10
CMSWest.Expiration = 20 * MinuteInSecs
# Icy Curves Warning

if SurfaceIceWarning or SurfaceIceWatch or SurfaceFrost or SignTest:
    CMSEast.logmsg = "Icy Road Warning. Values: %d, %d, %d, %d, %d, %d" % (RWIS.essSurfaceStatus1(),
    RWIS.essSurfaceStatus2(),
    RWIS.essSurfaceStatus3(),
    RWIS.essSurfaceStatus4(),
    RWIS.essSurfaceStatus5(),
    RWIS.essSurfaceStatus6())

    CMSEast.MessagePage1Line1 = "CAUTION"
    CMSEast.MessagePage1Line2 = "ICY ROAD"
    CMSEast.AddMessageToQueue()

    CMSWest.logmsg = "Icy Road Warning. Values: %d, %d, %d, %d, %d, %d" % (RWIS.essSurfaceStatus1(),
    RWIS.essSurfaceStatus2(),
    RWIS.essSurfaceStatus3(),
    RWIS.essSurfaceStatus4(),
    RWIS.essSurfaceStatus5(),
    RWIS.essSurfaceStatus6())

    CMSWest.MessagePage1Line1 = "CAUTION"
    CMSWest.MessagePage1Line2 = "ICY ROAD"
    CMSWest.AddMessageToQueue()

# Default Case - No Icy Curves Warning

if not SurfaceIceWarning and not SurfaceIceWatch and not SurfaceFrost:
    CMSlogger.info("No Icy Curve Warning")
Testing/Evaluation

• In-house
  – Individual modules tested to verify communications, data parsing, etc.
  – Integration Testing
  – Dial in to Fredonyer Pass RWIS to get real data from real RWIS

• Pilot Field Testing
  – In lab at Caltrans D2
  – Field Site – Spring Garden
Evaluation

• Technical Performance
  – Log and data files examined to evaluate correctness

• Reliability
  – Long term testing in lab and at Spring Garden

• Usability
  – Survey sent to Ken Beals of Caltrans District 2 concerning ease of system setup and administration
Evaluation cont.

• Maintainability
  – Autonomous system, goal to require little maintenance
  – Tried to simplify as much as possible what maintenance tasks there were (scripts, config files, checking status and logfiles)

• Security
  – Minimize surface area
  – Only ssh should be necessary, http is optional
Automated Safety Warning Controller
Pilot Test

- Controller was pilot tested at Spring Garden
- Pilot test was limited to RWIS and CMS field elements
- Pilot test was limited to Ice Warning alert script
- Device was installed by Ken Beals of Caltrans District 2
  - Installation of field elements
  - Network configuration
  - Alert scripts
Pilot Test

Fixes and Changes

• Added a sign test option to the front panel interface at Ken’s request

• CMS DisplayTime was off by a factor of 10

• Python logging module has a bug if more than one rotate interval passes between messages being logged

• CMS messages were logged when placed on sign and when deleted. A subtle timing issue would make the Controller think a different message was on the sign than there was.

• There was an issue with archiving of the data files. Files are archived on read, but the ice warning script only reads from memory, so the RWIS file wasn’t being archived.
General lessons learned

• Python as a language?
  – Good for rapid prototyping
  – A lot of functionality built in to the language

• Flat files vs. SQLite
  – SQLite is faster than just reading a flat file, but with some buffering techniques flat files become much faster

• Buffering data in memory vs. reading files
  – Managing buffers is complex
  – Offers significant speed improvements
Automated Safety Warning Controller
Phase 2

Kickoff meeting conducted May 25th, 2010 at Caltrans District 2 headquarters

- Outcome of Phase I of this project was a prototype hardware and software system that is being tested in the field

- Outcome of Phase II will be a hardware and software system that has been pilot tested by multiple users in the field and has been prepared for wider deployment

- Additional research and development will be conducted to prepare for deployment including fault tolerance of system software, production of training materials and other documentation, and preparation of business case material to assist in deployment justification
Automated Safety Warning Controller

Acknowledgements

Ken Beals – Caltrans District 2 ITS Engineer
Shaowei Wang, Sean Graham, Justin Krohn and others from WTI
Others …
Questions?
More information and future updates can be found at www.westernstates.org