INSTRUMENTATION AND EVALUATION OF DISTRICT 10 CALTRANS AUTOMATED WARNING SYSTEM (CAWS)

Technical Deliverables

Prepared for the California Dept. of Transportation and California Office of Traffic Safety by Loragen Corporation, San Luis Obispo, California

Project Manager: Andrew Lee, Caltrans Division of Innovation and Research

Principal Investigator: Art MacCarley

OTS Grant No. RS0034, Caltrans Contract No. 51A0050

Document No. L-D10-FR-05

September 25, 2005

Table of Contents

Tables

Table of Figures

Technical Deliverables

5.1 Background

This document describes the design, operation and maintenance of specialized instrumentation for the acquisition of detailed traffic flow information, deployed as part of the evaluation of the ***Caltrans Automated Warning System (CAWS)*** located in and maintained by Caltrans District 10. The ***CAWS Evaluation System*** described herein is a core project deliverable, and was required to provide the necessary field data to support the Driver Response Analysis and the Technical and Operational Assessment Tasks of this evaluation contract. This document is based on previously delivered Document No. OM1-D10-4/05, the *CAWS Evaluation System Operators Manual*.

The CAWS Evaluation System consists of hardware and software components located at five field locations on the CAWS highways, as well as at the Caltrans District 10 Traffic Management Center (TMC) and at Loragen Corporation in San Luis Obispo, California. A complete description of the CAWS and the multi-objective evaluation is described in other Volumes of this Final Report.

The CAWS Evaluation System is an adjunct to the CAWS itself, providing a means for assessing the effectiveness of this automated driver-warning network, located on Interstate Highway 5 and State Route 120 near Stockton, California. The CAWS was designed and deployed by the California Department of Transportation (Caltrans), and has been operating since November 1996. Below is a brief description of the CAWS at a level necessary to make this a stand-alone document.

The system includes 36 traffic speed-monitoring sites, 9 complete remote meteorological stations including visibility detectors, and nine changeable message signs (CMS) for warning drivers. The system is controlled by a network of three computers located physically in the District 10 Transportation Management Center (TMC), running specialized software developed by Caltrans Operations and the weather system vendor. This system is believed to be one of the most sophisticated of its kind in the world.

The evaluation project was comprised of four levels of assessment: long term effects on traffic accidents, influence of CAWS on driver behavior; operational/functional verification and assessment, and a technical assessment of deployed hardware and software components.

The driver behavior analysis most directly answers questions of cause and effect by directly observing and interpreting the reaction of drivers to warning messages displayed by the CAWS. The ultimate objective of the evaluation is to provide an objective basis for decisions in other jurisdictions to replicate (or not) the system.

We assess driver behavior by study of the behavior of individual vehicles as elements in a traffic flow, immediately before and after drivers have been exposed to a message on a CMS warning of an impending visibility or traffic problem. Driver behavior is characterized in terms of a number of metrics of relative

traffic safety, including but not limited to the first and second statistical moments of vehicle speed and separation.

A “control” test site was required for analysis of the traffic flow prior to exposure to the CMS warning message. No such site was originally included in the CAWS network. We selected two appropriate control sites on Southbound I-5 near the Downing Road onramp and adjacent to the French Camp Slough under crossing, both immediately prior to the north end of the CAWS study area. Power was routed to both sites, although landline data communications was not possible. Duplex inductive loop detectors were installed in each lane at each site for measurement of the speed and time of arrival of each vehicle. A visibility sensor was installed at the French Camp Slough site, the detector identical in capability to that deployed at the nine existing weather stations on the CAWS.

Two “after” sites were situated at Weather Station One, located on I-5 near Mathews Road under crossing, and at the site of traffic monitoring station 1B, below the El Dorado Road overcrossing. Remote data acquisition equipment was installed at all sites to record and telemeter data to a secure evaluation server via the internet. Additional data acquisition equipment was co-located with the CMS controller at the site of CMS No., the midpoint between the four evaluation monitoring sites.

We refer to the field sites, in north to south order, as:

Downing Road

French Camp Slough

CMS

Mathews Road

El Dorado Overcrossing

Figure 5.2.1.1 shows the location of the field monitoring sites overlaid on the map of the CAWS study area.

5.2 Caltrans Automated Warning System Description

The primary function of the CAWS is to detect the presence of reduced visibility and/or congested traffic on the highway, and to warn drivers in advance of such conditions. Particular emphasis is on combinations of these conditions: slow or stopped traffic ahead which drivers might otherwise not be aware of due to reduced visibility.

The CAWS is comprised of three primary subsystems:

5.2.1 Qualimetrics Caltrans Meteorological System (QCMS or Weather System)

The weather monitoring components of the CAWS were manufactured and installed by Qualimetrics Inc. of Sacramento California (now All-Weather Inc.) Nine remote weather monitoring stations are deployed on Highways 5 and 120, each including a dual axis atmospheric visibility sensor, an anemometer, barometer, thermometer, dew point sensor, precipitation gauge, and a telemetry system for encoding all instrument data and transmitting to a central weather monitoring computer. Data is carried over a network of

dedicated and leased telephone lines to the Traffic Management Center (TMC) located in Caltrans District 10 Headquarters in central Stockton. Information is displayed via a proprietary program running under Windows 95 on a PC in the TMC. Data is retained on disk until disk capacity is exceeded, when it must be manually backed up and erased. The weather monitoring PC has the ability to generate user-settable "alarms" when instrument ranges exceed threshold values. Of importance to the CAWS are fog (visibility plus relative humidity) alarms set at three levels: 500, 200 and 100 ft. (FAA visibility distances). An alarm threshold is also enabled for high wind conditions. The Weather Monitoring PC has an RS-232 link in the TMC to the Signview computer, which acts upon the alarm level flags passed to it, automatically activating visibility or wind-related warning messages according to a re-configurable look-up table.

Evaluation of Caltrans Automated Warning System

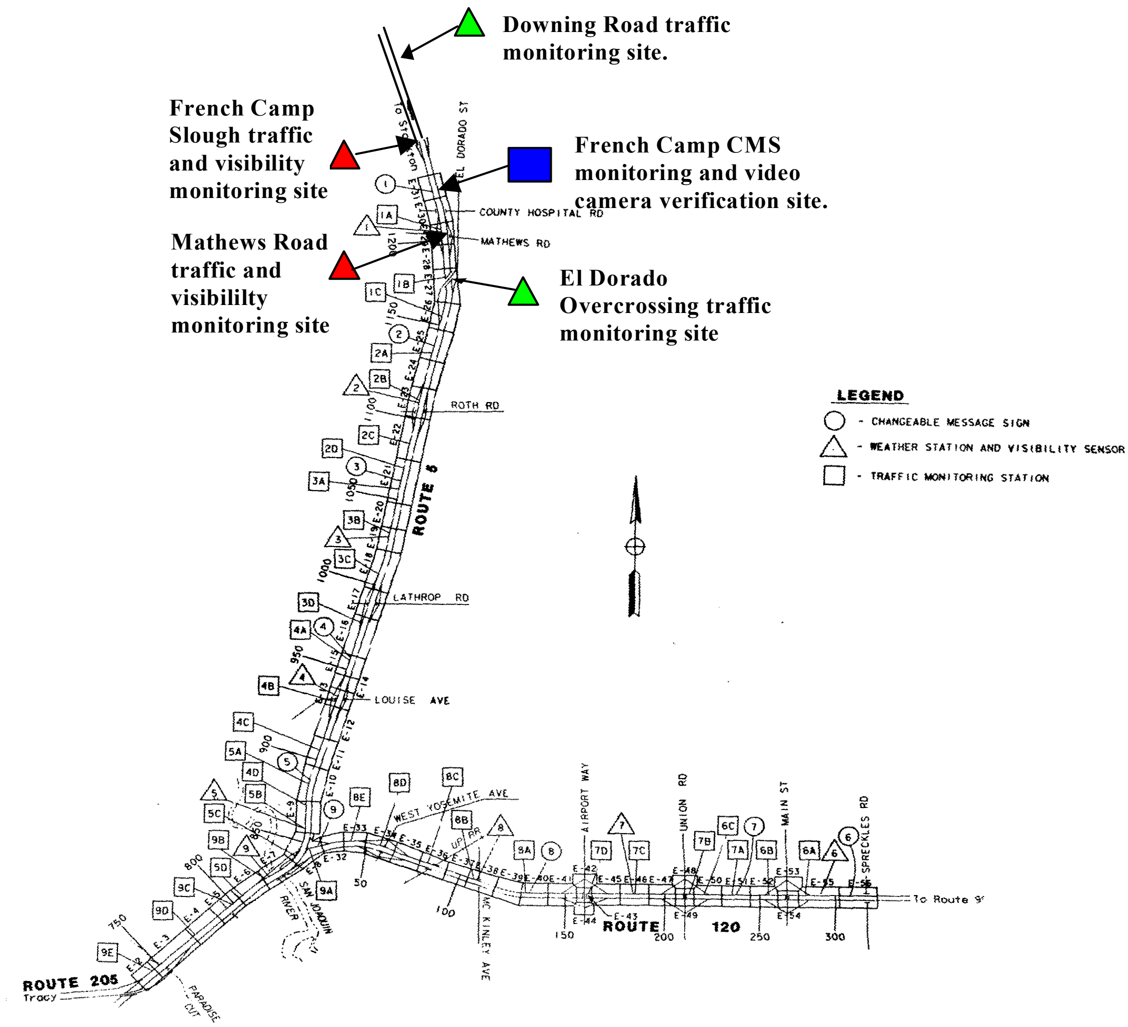


Figure 5.2.1.1. CAWS study area on I-5 and SR-120 near Stockton California, showing location of CAWS Evaluation Test Sites.

5.2.2 Traffic Monitoring System (TMS)

Duplex loop-pair speed detectors are installed at 36 sites, spaced at approximately 1/2-mile intervals on Highways 5 and 120 in the Stockton-Manteca area. Some are located at or proximate to weather stations. Six sites are designated as communications hubs, to which all other sites are connected via twisted pair in a “star-type” network. All installations conform generally to Caltrans and US DOT FHWA specifications and practices for field control systems[i]. Field interfaces are implemented via Type 222 dual-channel inductive loop detector cards. Traffic count and speed measurement algorithms are run on Caltrans Type 170 controllers, sourced by various venders. Data from each of the five communications hub sites are brought into the District 10 TMC over dedicated and leased telephone lines using Caltrans Type 400 (Bell L202S) 1200 bps modems at each end. Communications is full-duplex and a polling cycle of 30 seconds is used to retrieve data from all sites. Traffic count and speed data are displayed via the Caltrans-developed Traffic Monitoring System (TMS) program running under DOS on a PC in the District 10 TMC. The program has display capabilities for up to 36 sites. Reported mean speed data are averaged over the polling interval. Data is retained on disk until disk capacity is exceeded, when it must be manually backed up and erased. The PC communicates with the CMS-control (Signview) PC described below via an RS232c cable at 9600 bps.

5.2.3 CMS and “Signview” CMS Control System

Caltrans Model 500[ii] incandescent changeable message signs (CMS) are installed at nine locations in the study area. Incandescent-type (self-illuminated) displays have been reported to have the best readability under adverse weather conditions[iii]. All display sites are proximate to loop detector sites, and have dedicated Caltrans Type 170 controllers for display and communications. Communications is handled over dedicated and leased phone lines and Caltrans Type 400 (Bell L202S) 1200 bps modems, connecting the signs in star-type clusters to a single central monitoring PC in the District 10 TMC. Proprietary “SignView” software, developed by Caltrans staff, actives the displays, selecting warning messages based upon programmed levels thresholds of traffic speed, visibility and wind speed. The following fixed messages are currently deployed automatically:

- 2 traffic speed warnings: “SLOW TRAFFIC AHEAD/CAUTION” and “STOPPED TRAFFIC AHEAD/CAUTION” based upon thresholds of below 35 mph, and below 11 mph respectively.
- The fog-related messages have been changed several times since 1996. Originally “FOGGY CONDITION AHEAD” or “DENSE FOG AHEAD”, changed in January 2003 to “DENSE FOG AHEAD, ADVISE 45 MPH”, or for visibility between 100 and 200 ft, “DENSE FOG AHEAD, ADVISE 30 MPH”. Changed in summer 2004 to “DENSE FOG AHEAD, ADVISE 45 MPH” for all visibility levels less than 500 ft.
- 1 wind speed warning: “GUSTY WIND WARNING” is displayed when local wind speed exceeds 25 mph.

Priority-based preemption is implemented, with traffic speed-related warnings superceding visibility and wind-related warnings. Manually entered warning messages can be displayed at any time by operator override. All automated messages supercede manually placed messages. The Signview system receives inputs via EIA232c serial links from the adjacent weather monitoring and the TMS traffic monitoring PCs. The system is only accessible from the TMC. It cannot be remotely accessed via the Internet, the Caltrans

Intranet or a telephone modem. Figure 5.2.3.1 is a block diagram showing the three main computer subsystems of the CAWS, and their respective communications with field sites and sensors.

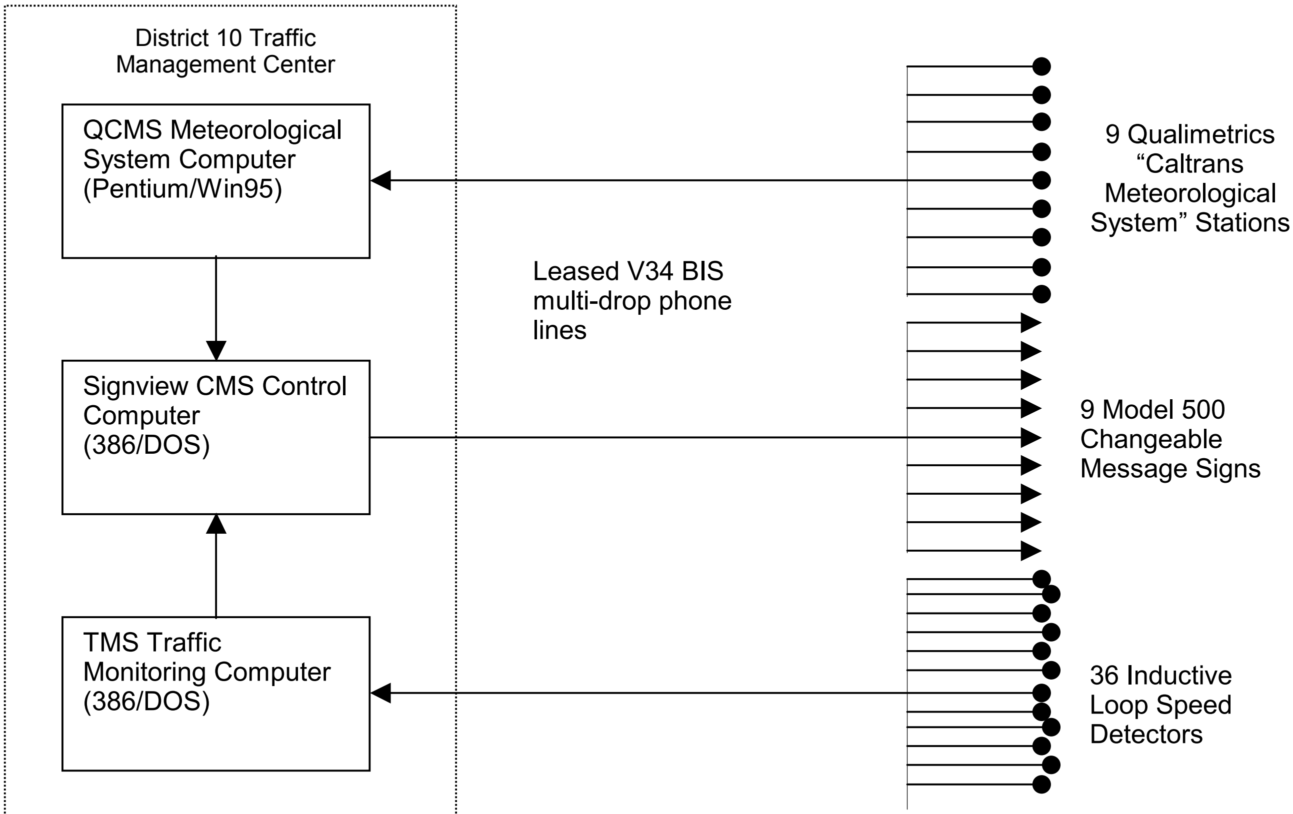


Figure 5.2.3.1. Main Components of CAWS as deployed in Caltrans District 10.

The CAWS has capabilities beyond those currently in use for automated driver warning. Each of the nine weather stations, described in Figure 5.2.3.2 includes a full complement of atmospheric monitoring instrumentation. The central weather monitoring PC incorporates comprehensive data collection and interpretation capabilities. Combined with the network of 36 speed detection sites and its associated central traffic monitoring computer, additional reporting, alert generation, and data collection functions are possible. As a test bed for weather-related intelligent traffic management, this system appears to be one of the best equipped in the United States, and possibly the world. Please refer to the Bibliography section of the CAWS Final Evaluation Report information on similar systems in the US and throughout the world.

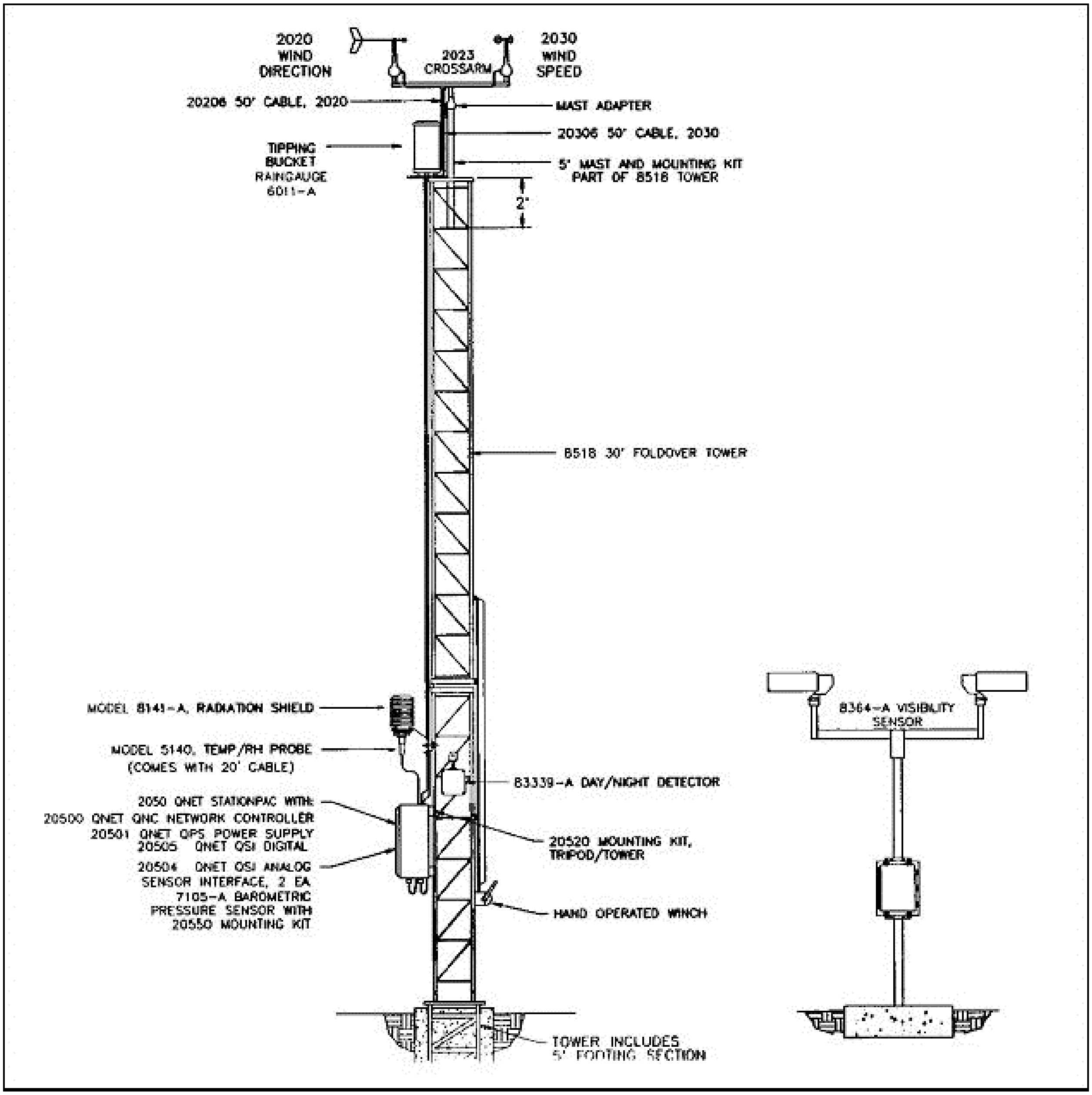


Figure 5.2.3.2. Qualimetrics Remote Weather Monitoring Station as deployed in CAWS.

Figure 5.2.3.3 from a Caltrans internal presentation by Joel Retanan, Floyd Workmon and Celso Izquierdo of Caltrans operations, illustrates the role of each key component of the CAWS.

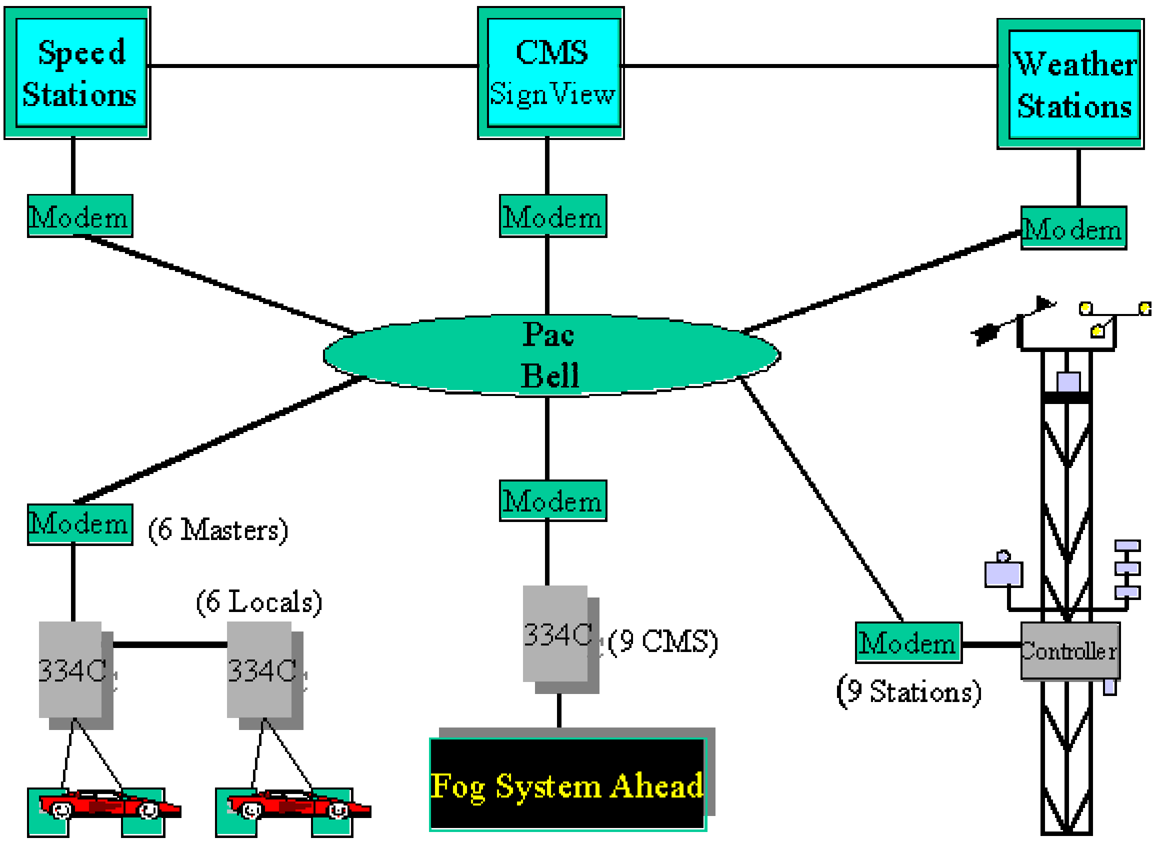


Figure 5.2.3.3. CAWS key components and interconnectivity (provided by Caltrans Operations).

[i] US Department of Transportation. “Traffic Control Systems Handbook” FHWA-IP-85-11, Chapter 5 *Detectors* and Chapter 8 *Communications*. April 1985.

[ii] California Business, Transportation & Housing Agency, Department of Transportation. “Specifications For Changeable Message Sign Systems”, Chapter 2. Available from Caltrans Division of Traffic Operations, 1120 N Street, Sacramento, CA 95814, January 1992.

[iii] Pouliot, Stephen G., and Wilson, Eugene M. “Motorist Information Needs and Changeable Sign Messages for Adverse Winter Travel.” Presented at *Improving Rural Transportation Through Advanced Transportation Technology*, conducted by Caltrans and US Dept of Transportation FHWA, Redding, California, September 28-29, 1992.

5.3 CAWS Evaluation System Components

The CAWS evaluation system was designed primarily to enable the detailed assessment of driver behavior in reaction to warning messages displayed by the first CMS encountered by drivers entering the CAWS study area from the north. Figure 5.2.3.1 illustrates the key elements of the evaluation system.

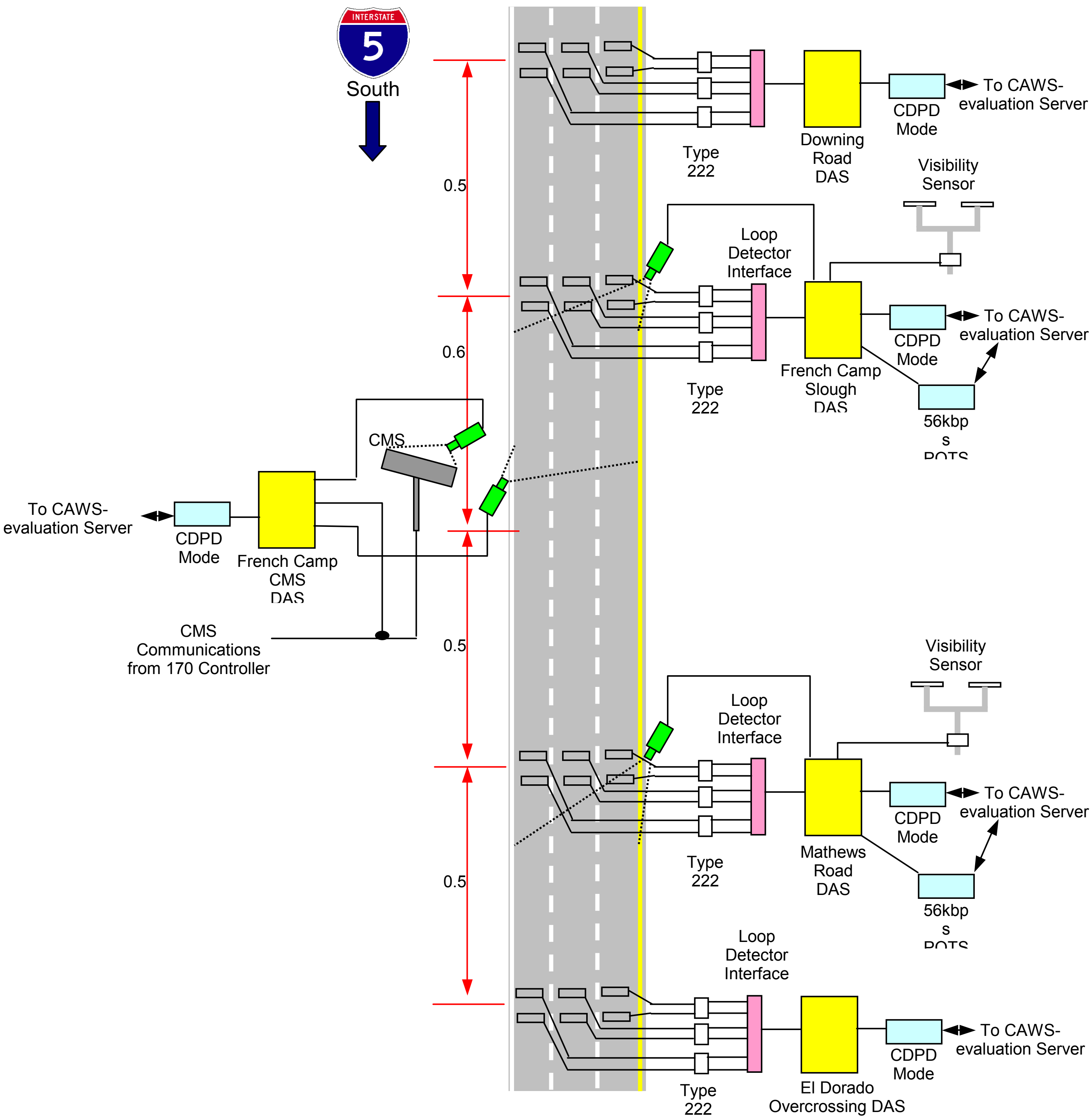


Figure 5.2.3.1. Functional Diagram of the Evaluation System.

The basic architecture of the CAWS Evaluation System is described in the block diagram of Figure 5.2.3.2 below. Complete details and the design philosophy behind this deployment are contained in the Final Project Report "Evaluation of the Caltrans Automated Warning System – Final Project Report", distributed by the California Department of Transportation and California Office of Traffic Safety, 2005.

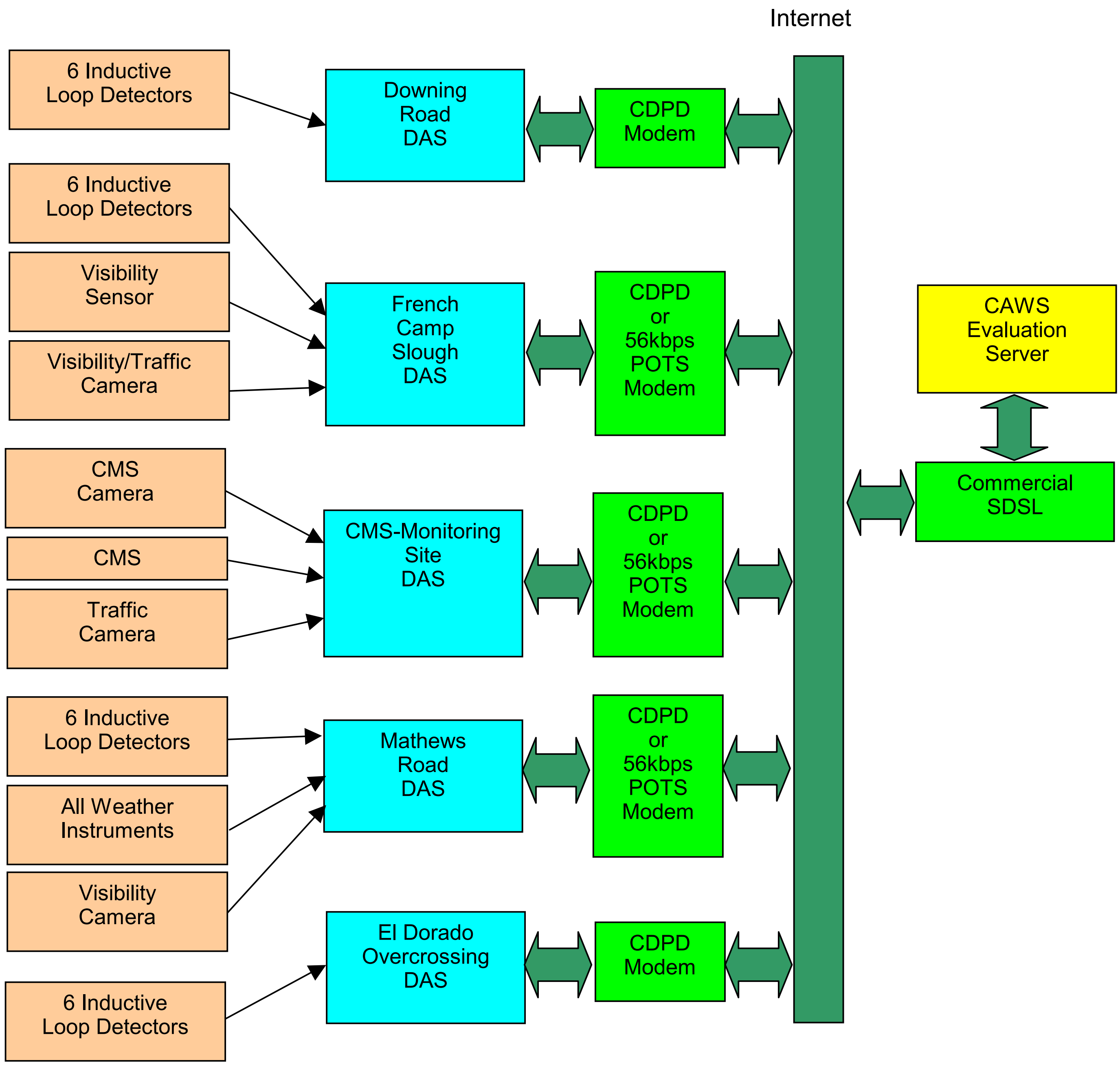


Figure 5.2.3.2. CAWS Evaluation System Architecture.

The symbolic and block diagram of Figure 5.2.3.3 below depicts the instrumentation, comutational and communications components located at the two inner-most (before CAM and after CMS) monitoring sites.

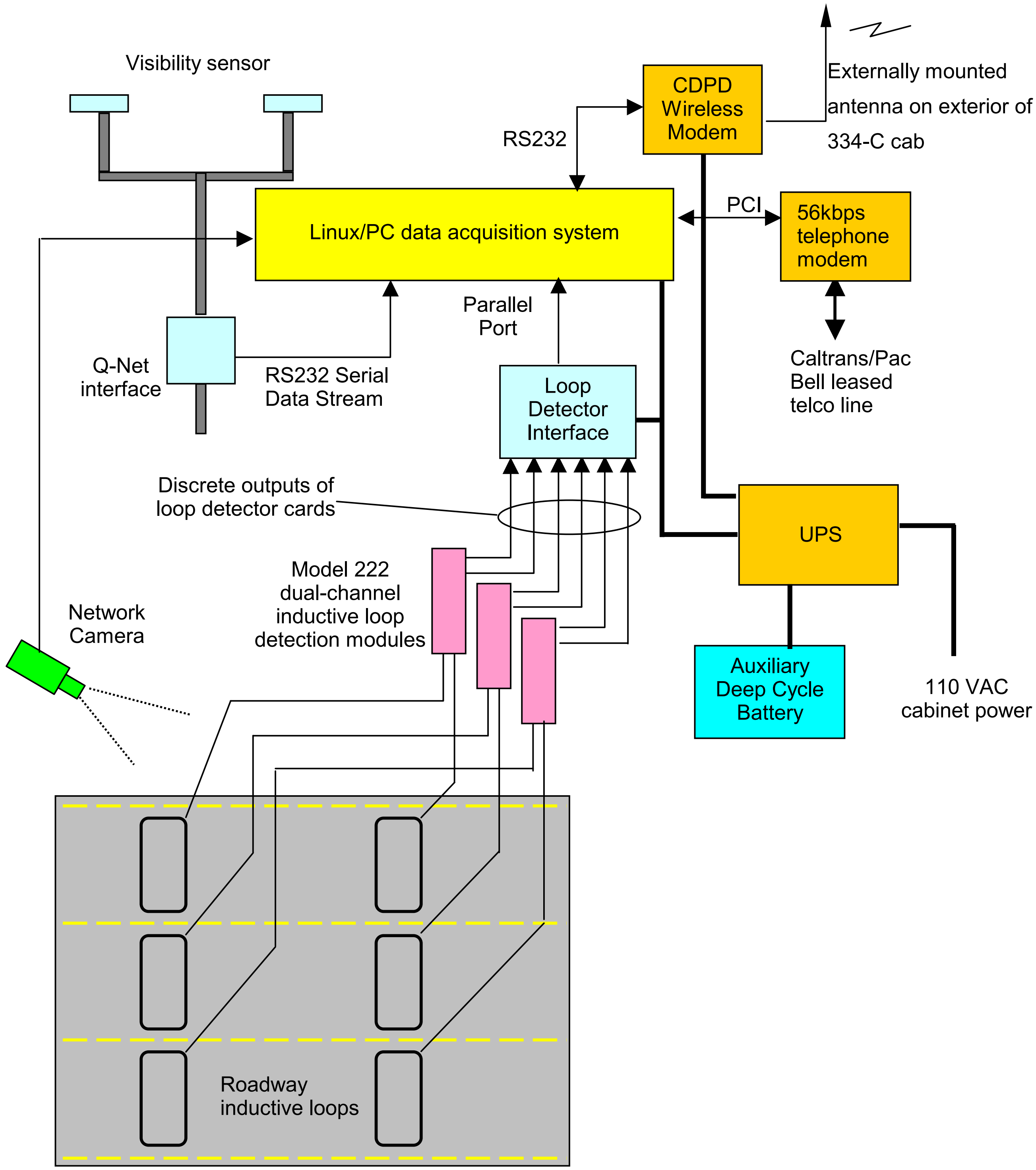


Figure 5.2.3.3. Data Acquisition System, Two Sites: French Camp Slough and Mathews Road.

A typical field computer installation in a Type 334-C cabinet is shown in Figure 5.2.3.4 below. The black unit at the top of the photograph is the CAWS field data acquisition system (DAS), a 4-unit rack-mount

industrial PC with filtered forced-air ventilation. Above the DAS is a Caltrans specification 170 controller, in this case for control of the CMS. Below the DAS are the loop detector card racks with powered backplane. In the rack, to the left, are three duplex loop detector cards. We have standardized on Sarasota GP6c detector cards, set for presence mode detection (not pulse mode detection). The sensitivity thresholds for these cards are generally set to level 2 or 3, which is lower than the typical settings (4-5) used by Caltrans technicians for pulse-mode detection sites. Below the 222 card rack is the power supply for the 170 and 222 card rack. The components at the bottom of the cabinet are the UPS and its auxiliary deep-cycle battery.

Figure 5.2.3.4. Typical installation of field data acquisition computer and UPS in Type 334-C cabinet.

At the CMS monitoring site located at the location of CMS No. 1, we deployed an identical DAS which is configured for monitoring communications between the 170 controller and the CMS, as wells as

supporting and transmitting images from two network video cameras: one for monitoring traffic and visibility conditions, and one for verifying the message actually displayed by the CMS. A block diagram of this specialized system is shown in Figure 5.2.3.5 below.

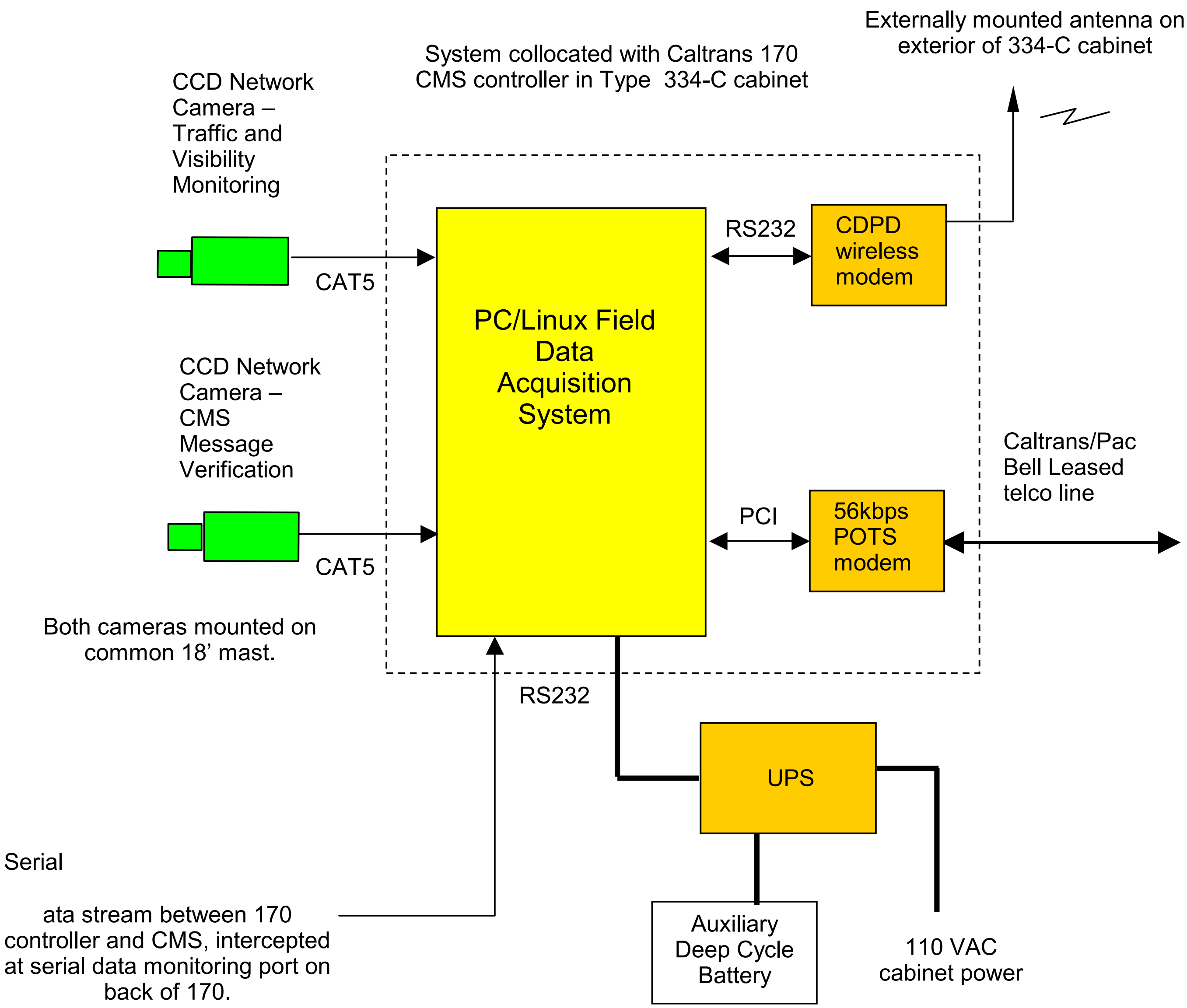


Figure 5.2.3.5. Video monitoring system located immediately upstream of CMS No. 1 for verification of current CMS message and confirmation of fog severity.

Figure 5.2.3.6 shows the CMS and traffic monitoring cameras installed at CMS site No. 1. The data acquisition computer and related components are co-located inside the open Type 334-C cabinet.

Figure 5.2.3.6. CMS and traffic monitoring cameras at CAWS CMS No. 1.

Figure 5.2.3.7 below illustates the equipment deployment at the two outer-most field sites, at El Dorado Overcrossing and Downing Road. These sites collect only traffic data, communicate only with wireless modems, and are not equipped with visibility sensors or cameras.

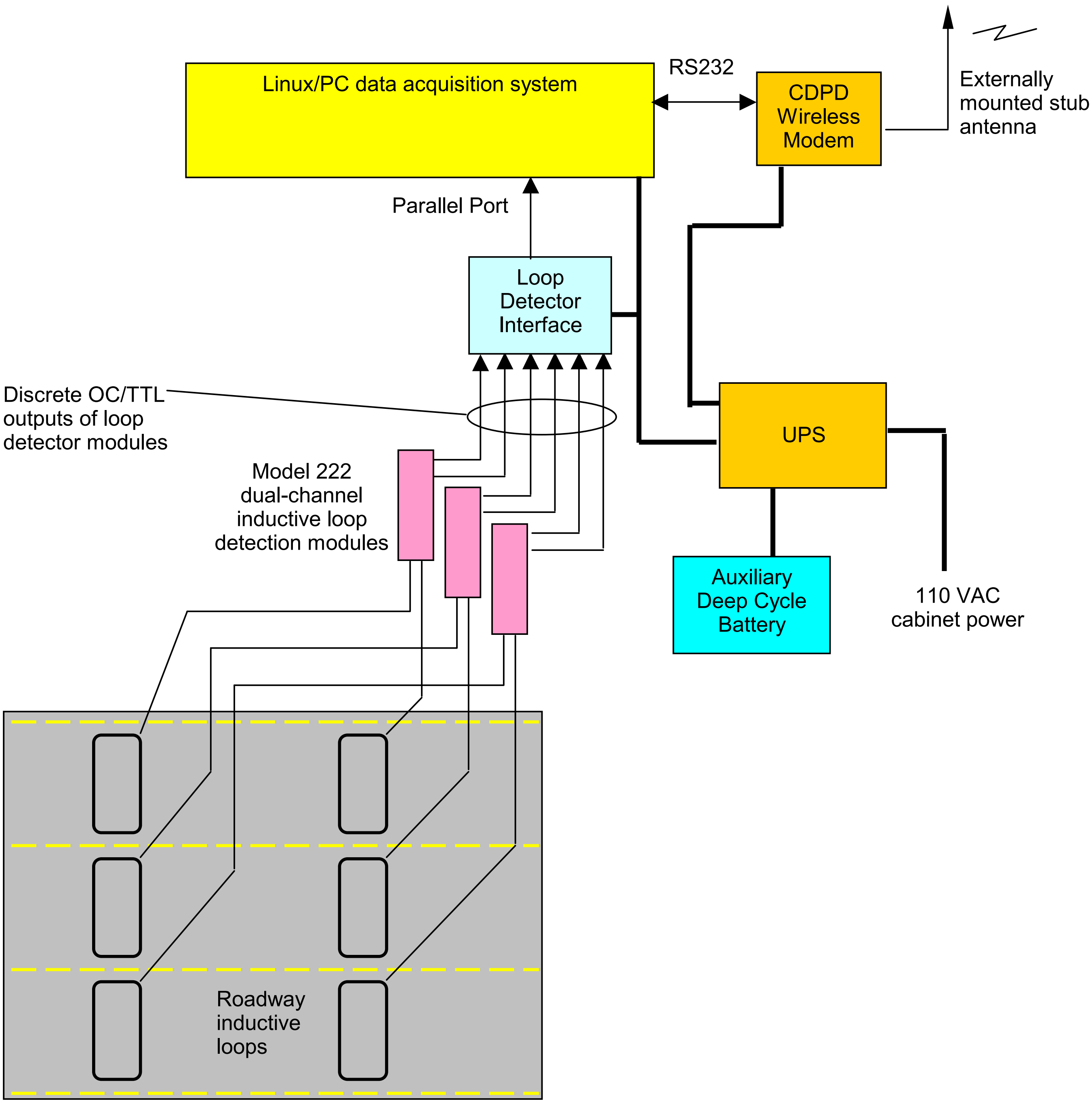


Figure 5.2.3.7. Data Acquisition System Two Sites: El Dorado and Downing Road Sites.

Figure 5.2.3.8 shows a screen capture of the public front page of the CAWS evaluation web site caws-evaluation.loragen.com. Page includes a dynamic display of the current traffic, visibility and image data from the field sites. This page also serves as the entry point for authorized access to detailed data and analysis tools on sub-pages.

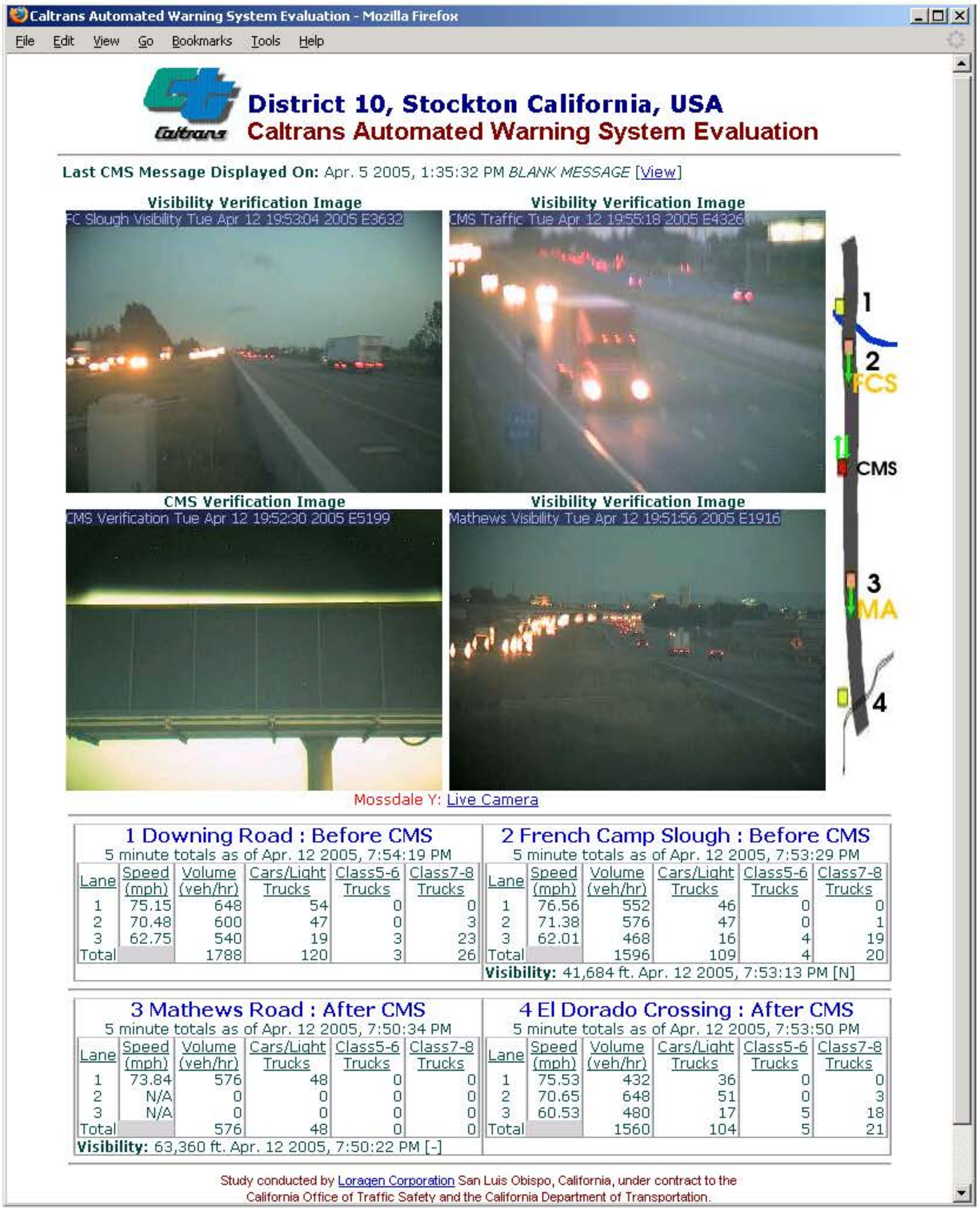


Figure 5.2.3.8. Screen capture of CAWS Evaluation Web Sites.

5.4 Field Systems

5.4.1 Data Acquisition System Hardware

The Evaluation Data Acquisition System (DAS) computer is a PC-type platform constructed of high-reliability components in an industrial 19” rack-mount enclosure. All components are commercial off-the-shelf. The 4-U enclosure includes high-flow filtered forced are cooling, but is otherwise sealed to the environment. Each of the six DAS computers (five in the field and one as a spare) contains identical hardware, to assure that they are interchangeable with only software re-configuration. Each can be substituted for each other with minimal software modifications (see section 5.11.7 on how to swap field units)

Table 5.4.1. Hardware list for each field computer (DAS).

	Description and Configuration
Rackmount Computer Case	19” 4-U Industrial PC with 400W high-reliability power supply. S411/RA455A Model, IPC 4U Series http://www.uslogic.com/igc/code/igcrack.html
Motherboard	Abit VH6-II (Abit Computer Corp. San Jose, CA.) http://www.abit-usa.com/ Temperature monitoring driver: [lmsensors via82C] http://secure.netroedge.com/~lm78/via686a-isa-6000 Modifications: 5V power on parallel port pin Note: Bend temperature sensor down so it doesn't touch CPU
RAM	1024MB (1GB) PC100 SDRAM Micron
Flash Memory Drive	M-Systems IDE 3000 2.5” 256MB Flash Drive #IDE-25-256-X configure: Primary IDE Controller as Master Add 44 pin IDE to 40 pin IDE + Power Adapter Mount inside the case. product: http://www.m-sys.com
Hard Drive Exchangeable Rack	Lian Li RH-17 Mobile Rack for 2.5” & 3.5” Drive ATA133 product: http://www.lian-li.com

	Description and Configuration
Hard Drive Exchangeable drive module	Lian Li RH-17 product: http://www.lian-li.com
Hard Disk Drive:	Western Digital Caviar 20 GB Hard Disk WD200BB LBA 30102336 configure: Secondary IDE Controller as Master (commodity component)
CDROM	Liteon 52X Maximum configure: Secondary IDE Controller as Slave (commodity component)
Floppy Disk Drive	Standard 1.44 MB 3.5" floppy drive NEC or Mitsumi configure: floppy drive A: (commodity component)
Processor	Pentium III 1GHz CPU running at 750MHz 100MHz bus (commodity component)
CPU Fan	Speeze CPU Cooler and Fan (passive cooling) (commodity component)
AGP Video Card	Video on motherboard using Jaton Video-107AGP AGP 2x (Trident Blade Chipset)
Network Card	CAT5 100 base-T network on motherboard, using RTL 8139C Chipset driver: RTL 8139C Realtek Generic
Dial-up Modem Card	BM 33L4618 V.90 PCI Data/Fax Modem driver: kernel built-in auto-detect as /dev/ttyS4 (commodity component)
Watchdog Timer Card	Berkshire Watchdog PCI card configure: hardware jumpers DDDDUUDD product: http://www.berkprod.com driver: http://www.pcwd.de

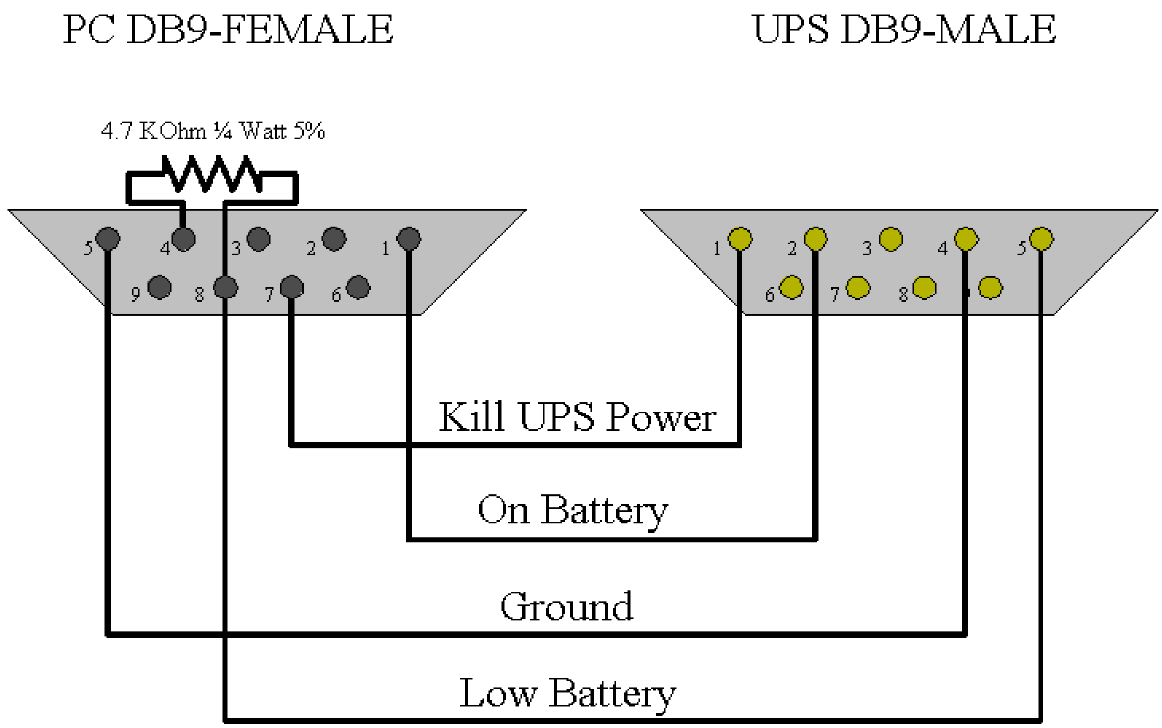
	Description and Configuration
ISA Serial Port Card	Startech.com 1 Port ISA 16C550 Serial Card #ISA1S550 configure: hardware jumpers COM3, IRQ4, 115.2K bps product: http://www.startech.com driver: kernel built-in auto-detect as /dev/ttyS2

5.4.2 UPS

An Uninterruptible Power Supply (UPS) is installed at all field sites, to provide backup power in the event of a utility power failure. All units are APS (American Power Systems) 650 Watt “BackupsPro 650” models. Each UPS has been physically modified to connect to an auxiliary deep cycle battery (Marine/RV Group 24 or 27). This increases the time that a system can run on battery power, Tests have shown that these modified UPS’s can supply enough power to keep a computer system running for approximately 3 hours. After typically 3 hours have elapsed, the UPS will issue a low-battery condition to the computer. This signal facilitates the computer shutting down the UPS to prevent damage to the both itself and the UPS. We have observed that it is possible for the computer to continue to run with the low-battery signal on the UPS for as long as 2 hours, but elected the most conservative approach.

A special serial port cable is required to interface the UPS to the computer. This special serial port cable connects directly to a COM port on the back of a computer. The APC BackupsPro 650 model UPS uses “simple” signaling. This means that the RS232 signal lines are used to carry the UPS signals directly and that no RS232 protocol or communication is ever established between the PC and the UPS.

APC designed the direct signaling interface of the UPS and the manufacturer-provided cable to work with Microsoft Windows operating systems. The Microsoft Windows Operating System only provides direct serial port line access to 2 lines. This allows Microsoft Windows computers to only detect if a UPS is on battery power or line power. Our Linux system, however, has direct access to **all** the serial port control lines. Because the UPS itself supplies multiple control lines, more functionality can be obtained by creating custom UPS cables. To provide these enhanced features, we fabricated a custom UPS cable, specified below:



Note that the "Low Battery" signal is active low and must be pulled up using port pin #4 on the PC side (DTR). Note that the "On Battery" signal is also active low but is pulled up to the proper voltage inside the UPS.

This cable allows the computer to detect when the computer is on battery and when the battery is low. Once a low battery condition is detected, the computer can shutdown all its processes and then power itself off by sending a kill signal to the UPS.

New custom cables must be tested thoroughly because correct system operation during power outages depends upon the cable for reading the signal lines as stated above.

The field computers are equipped with an additional IO Port ISA Slot card that contains the correct COM port for the UPS. This COM port has its onboard jumpers configured for COM 3 & IRQ 4 & 115.2 Kbps. If a different IO card is installed in the computer, configure the jumpers of that card appropriately. Failure to do so could cause the system to have a conflict between the motherboard serial ports and the modem (set to COM 4).

5.4.3 Wireless Modems

The first phase of the evaluation required the Internet connectivity to only three field sites. At the time, the southern most sites (French Camp CMS and Mathews Road) were equipped with land based telephone lines. The northernmost site (French Camp Slough) did not have a telephone line installed. Communications via a wireless CDPD modem was the solution. Over the course of the first year, we found that the CDPD modem provided much more reliable communications than the Caltrans-leased

phone lines, so we installed CDPC modems at the other two sites also, and retained the copper lines only as a backup channel.

Another wireless option we considered was an 802.11 wireless Ethernet local area network. Since District 10 had already purchased a large number of Cisco wireless modems, routers and antennas, we were encouraged to look into this. However, since Caltrans never set up the wireless infrastructure, and would not have permitted external access foe security reasons, this option was not viable.

CDPD (Cellular Digital Packet Data) Modems are installed at the field sites. All modems and wireless data plans have been purchased from Airlink Communications (www.airlink.com). The wireless services are actually provided by AT&T wireless, resold through Airlink. AT&T guarantees the CDPD structure to be in place until June of 2005, the end of the evaluation contract. The site at French Camp Slough was the first site installed with a CDPD modem. The first modem available was the Raven CDPD. This modem has since been working successfully at French Camp Slough. The rest of the field sites use a newer CDPD modem version called the Redwing CDPD modem.

The CDPD modems can be configured via a utility called “Wireless Ace” available from the Airlink Communications web site. A straight through serial cable is used from a Microsoft Windows based computer to connect to the modem and configure its settings. Only a few changes are necessary from the standard configuration and these are detailed below. The most important change involves setting the BAUD rate to 19,200 (the CDPD modem's theoretical maximum throughput).

Table 5.4.2. CDPD modem settings and configuration data.

	New Value
CDPD	
Device IP Address	xxx.xxx.xxx.xxx (from carrier)
Device Port	1
Side Preference	1: A side only
Service ID Preference	Don't Care
Service ID	0/0/0
Channel List Mode	2:Hot Channel List
Channel List	676,667,683,686,674,675,720,673,684,677,679,669,687,670,0,0,0,0,0,0,0,0,0,0,0,0
3 Watt Booster Support	0:No Booster Attached
Disable Side Switch	0:Switch back to preferred side (default)
Serial Debug Output	0:No Serial Debug Output
Sleep Mode	0
Connection	

TCP Auto Answer Mode	0: Disable TCP Server
TCP Establishment Timeout	45
Data Forwarding Timeout	1
Data Forwarding Character	0
Destination IP Address	10.0.10.0
Destination TCP/UDP Port	12345
Destination Connect Mode	T
AT Command Compatibility	1:Standard Modem Compatibility
Ignore DTR	0:Use DTR
Startup Mode Default	0:AT Startup Mode (normal)
UDP Mode Default	0:Normal UDP
Telnet Echo Mode	0:No Telnet Echo
UDP Half Open Mode	0:Disable UDP Half Open
UDP Half Open Mode Timeout	5
Allow Any UDP IP	1:Allow Any UDP IP
UDP Half Open Response	0:No RING CONNECT
Break on TCP Connect	0
Delay Connect Response	0
Command Echo	1:AT Command Echo On
Command Response Mode	1:Verbose AT Responses
Quiet Mode	0:Output AT Result Code
Call Progress Result Mode	0
TCP Inactive Timeout	1
Specify TCPT in Seconds	0:TCPT Units are Minutes
Allow TCP Suspension	0:No TCP Suspension
Disable Reset on ATZ	0:Normal Reset(recommended)
Disable AT Esc Sequence	0:Enable AT Escape Sequence
Dial UDP Always	0:Disable (default)
COM Port	
Flow Control	0:No Flow Control
Baud Rate	19200
Data Bits	8
Parity	N
Stop Bits	1
Security	
AT Password	[none – use null string: “”]
Friends Mode	0:Allow Any
Friend List IP 0	
Friend List IP 1	
Friend List IP 2	

Friend List IP 3	
Friend List IP 4	
Friend List IP 5	
Friend List IP 6	
Friend List IP 7	
Friend List IP 8	
Friend List IP 9	

5.4.4 Land-line telephone modems

Two of the field sites are have copper leased telephone lines available: the CMS Monitoring Site and Mathews Road. For these sites a prioritized sequence is implemented by which the systems first make several attempts to connect via the wireless modems, and after successive failures, attempts to dial out via the land line 56kbps FLEX modems. This is implemented via the script described in Section 2.3.4. Local dial-up ISP contracts are maintained to support this option, which has only rarely been used.

5.4.5 Web Cameras

Two StarDot Netcam cameras are located at the CMS #1 site. One camera points toward the traffic and the other points towards the sign for message verification. The cameras are connected to a small local area network that is attached to the field computer. The diagram of the local 100-base-T/CAT 5 network is shown in Figure 5.4.5.1.

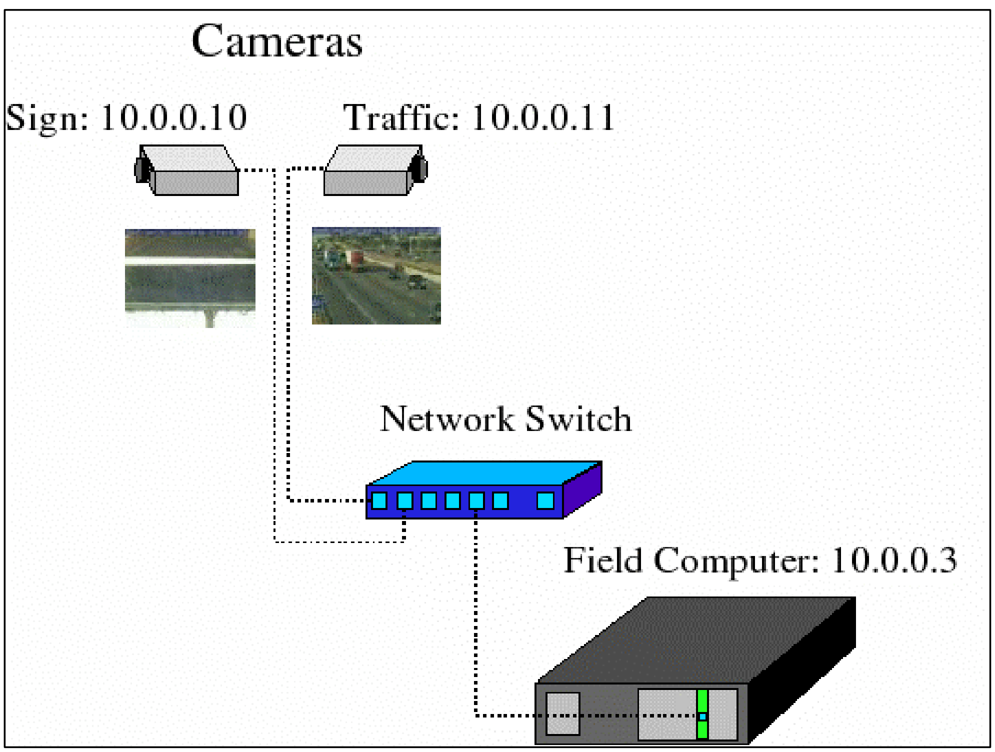


Figure 5.4.5.1. Network cameras at CMS monitoring site, with internal IP addresses.

Device local IP addresses:

Sign Camera	Local IP Address 10.0.0.10
Traffic Camera	Local IP Address 10.0.0.11

Remotely configuring the cameras from an Internet connected location is described below.

5.4.5.1 How to Access the Camera Configuration with SSH and Lynx

The first step involves logging into the French Camp CMS computer. Use SSH (Secure Shell) to connect to port 23 of the French Camp CMS computer's current IP address (166.130.0.18). The current IP address of each field computer can be found under the "Site Status" web page of the evaluation server.

[userprompt @ mycomputer]$ ssh loragen@166.130.0.18 -xC

-x connect to a computer without using an X terminal session

-C use compression when sending data over the SSH connection

loragen@166.130.0.18's password:

[loragen@loragen-field-3 loragen]$ lynx 10.0.0.10

The camera's internal configuration web page is only available through the field computer and not directly from the Internet. Lynx is used to access the camera's web page once an SSH (secure shell) connection has been made to the field computer. The Linux utility "lynx" is the most widely used text mode Internet browser.

Lynx will prompt for a user name and password when the camera's configuration link is activated. Enter in "admin" as the response to both prompts. This is the default user name and password for the camera. The user name and password for the camera's web page can be changed through the camera's web interface.

After entering the user name and password, a plain text view of the camera's web page can be viewed. A screen-shot of miscellaneous configuration options can be seen in Figure 5.4.5.3. Please refer to the camera's own documentation when modifying these settings.

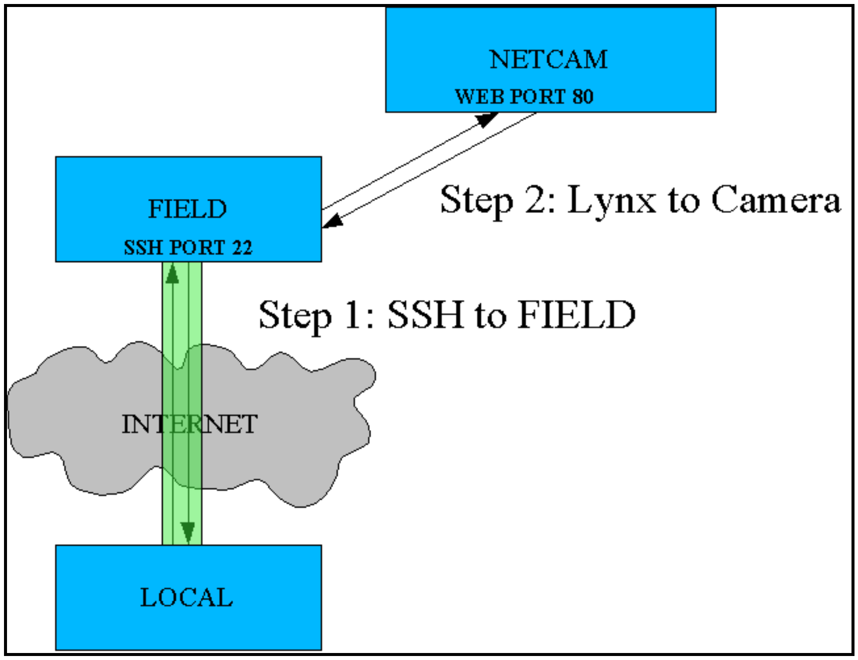


Figure 5.4.5.2. SSH Method 1.

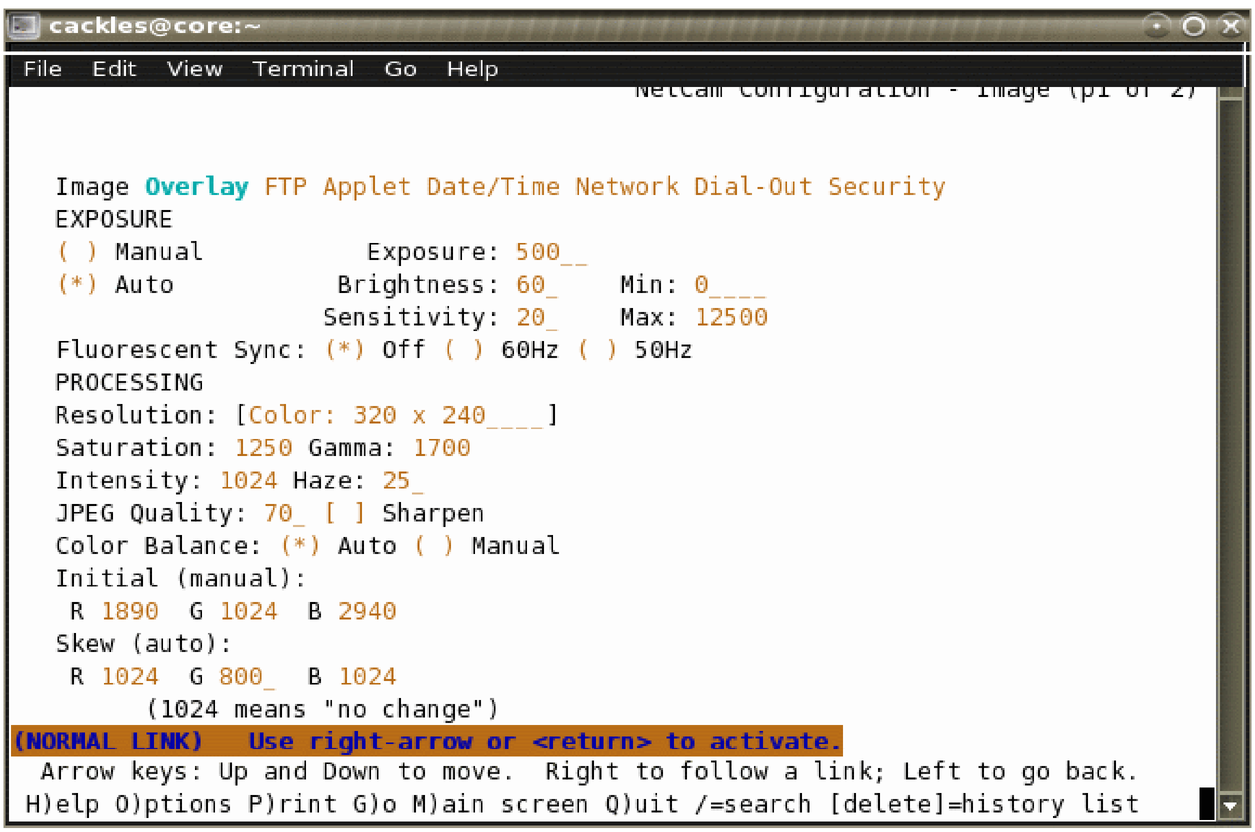


Figure 5.4.5.3. Lynx Text Browser.

5.4.5.2 How to Access the Camera Configuration with SSH Tunneling and Port Forwarding

Another method for accessing a NetCam's settings involves port tunneling or forwarding through an SSH client program. Essentially, the NetCam's web page can be viewed directly as if the NetCam was connected directly to the remote PC by tunneling the NetCam's web access port through the SSH encrypted connection. This process maps a local port of the local computer to a specific service at the other end of the SSH tunnel. All traffic is encrypted through the tunnel, so normal web page transfers appear to have additional communications overhead.

SSH tunneling provides a convenient method for access to the cameras' configuration settings, but requires more bandwidth than the previous method discussed (using SSH and Lynx). This is because Lynx does not download the images on the NetCam's web pages. Lynx only accesses the text content of the configuration pages.

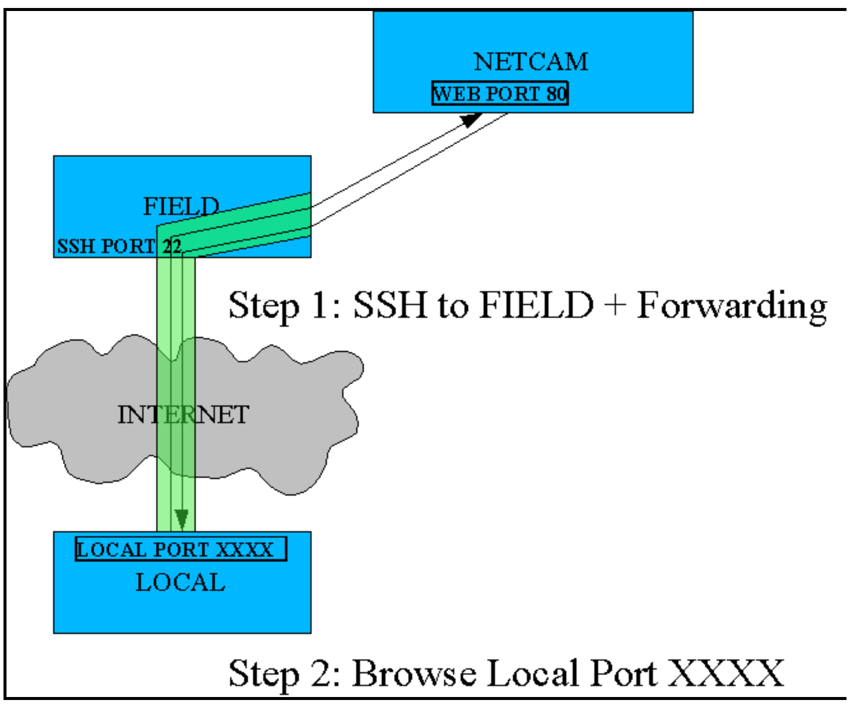


Figure 5.4.5.4. SSH Method 2.

[userprompt @ mycomputer]$ ssh loragen@166.130.0.18 -xC –L 2000:10.0.0.11:80

-x connect to a computer without using an X terminal session

-C use compression when sending data over the SSH connection

-L create a tunnel from port 2000 on local computer to port 80 of 10.0.0.11

loragen@166.130.0.18's password:

[loragen@loragen-field-3 loragen]$

Access to port 2000 on local computer will connect to the web server at 10.0.0.11 (the CMS traffic NetCam in the example). Visit the URL http://localhost:2000 to view the web page.

A tunnel to the NetCam's telnet server in order to control its FIZ (focus, iris, and zoom) capabilities. Add another tunnel to port 23 of the NetCam's IP address to access the NetCam's operating system with telnet.

Common SSH clients such as PUTTY and FSECURE™ require special configuration options to enable tunneling. If using one of these programs, please review the help files and search for "TUNNELING" in the supplied documentation.

Please visit http://www.openssh.com or http://www.ssh.com for more information about the secure shell protocol.

5.4.5.3 Camera Deployments at other Monitoring Sites

On December 14, 2004 two additional cameras were installed to verify the visibility sensor measurements at the evaluation sites. Cameras were placed at both the French Camp Slough and Mathews Road sites. These new cameras were oriented in the direction of southbound traffic. The images from these cameras present a similar view to that observed by the drivers heading into the CAWS area of influence.

The new cameras are connected similarly to those at the CMS site. At each site, a switch was installed to connect the field computer and the camera to the same network. The camera at the French Camp Slough site was assigned an IP address of **10.0.0.12**. The camera at the Mathews Road site was assigned an IP address of **10.0.0.13**.

One additional camera was deployed at weather station 9. Weather station 9 is the final weather station of the CAWS and overlooks the merge point of westbound SR-120 traffic with southbound I-5 traffic. The final camera serves to verify the visibility and traffic conditions at this critical location, at which a disproportionate number of accidents occur in the CAWS study area. This camera connects directly to the Internet through a wireless 819s CDMA modem from Land-Cellular. This is the only non-CDPD wireless site. Since the camera is directly accessible from the Internet, user and password information has been changed from the default NetCam settings and is included below.

MDN: 209-401-1938
MIN: 209-401-1938
ESN: 069-02508605
Accnt#: 770314886
SID# 00112

IP Address: 166.139.116.229

Ethernet port: 10.0.0.14

username and password: see Addendum A

This camera is set to automatically upload its images to the CAWS-evaluation server. The camera uploads an image to the FTP server every 5 minutes. This interval can be set through the camera's own internal configuration web pages. The upload interval may be optimized to fully utilize the bandwidth available over the CDMA wireless link. The NetCam's image can be directly accessed through the following web link: http://166.139.116.229/netcam.jpg (a username and password is required – see Addendum A).

The Ethernet port of this NetCam is currently disabled to allow the serial port modem to be the default connection to the Internet. Although it is not currently enabled, the NetCam's Ethernet port has been configured to use 10.0.0.14.

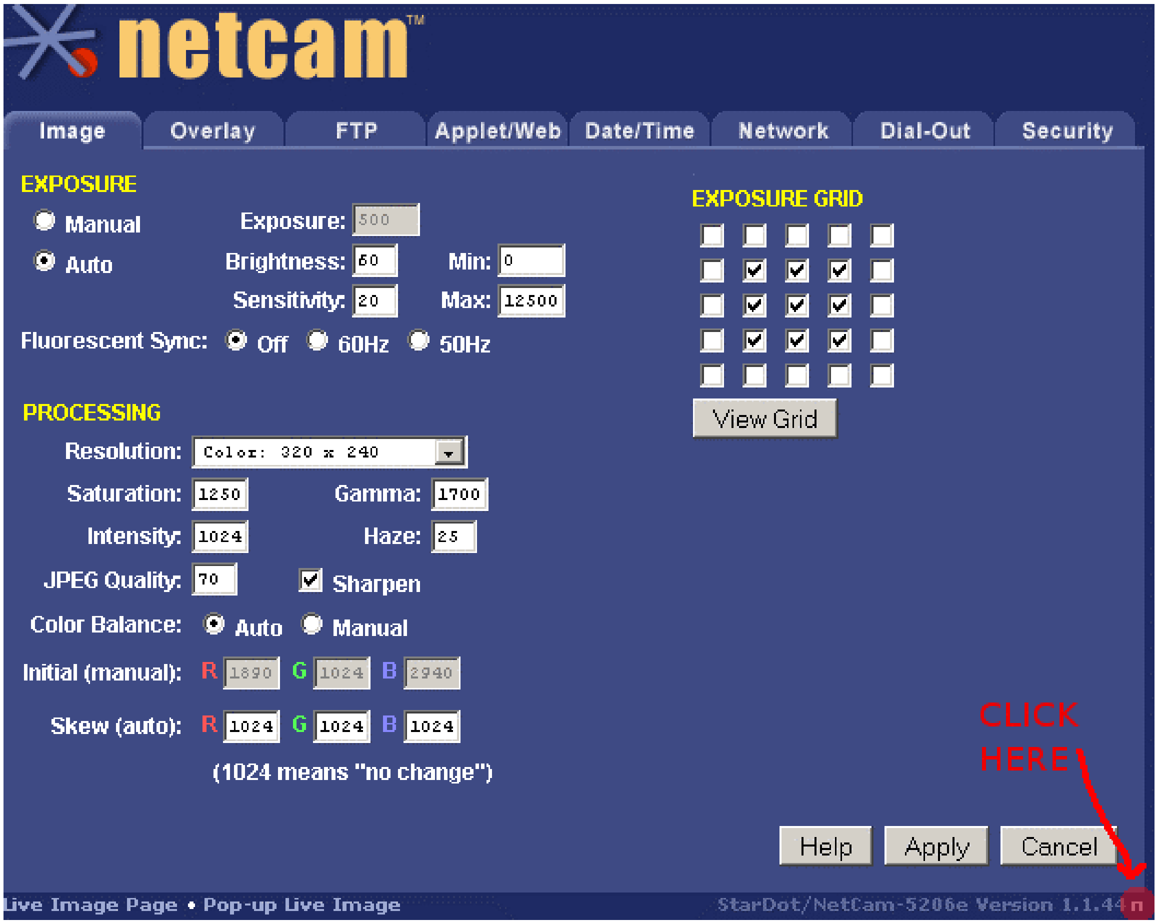
As of February 18, 2005 communication with this camera failed due to unknown errors. The NetCam and CDMA modem were removed from this site and are no longer installed.

The NetCam's interface and configuration settings for interoperability with the Land-Cellular CDMA modem are described below.

5.4.5.4 Enabling Stardot NetCam to work with LandCell CDMA CDM-819S Modem

To modify the configuration files of the NetCam, bring up the NetCam's web page and click on the configuration link.

Next, click on the small advanced settings icon. This icon is located in the very bottom right corner of the screen next to the firmware revision number.



After clicking the icon, the browser will be directed towards a page that lists many configuration files on the NetCam. This page gives the ability to edit and save the configuration files to flash memory. Make sure to always write configuration changes to flash memory. Otherwise, the NetCam will lose modified settings the next time it reboots (around 12:00AM in the morning).

Edit the files “/etc/config/chat.dialout”, “/etc/config/dialout.conf”, and “/etc/config/diald.conf” so that they are identical to the following configuration table.

/etc/config/chat.dialout	/etc/config/dialout.conf
TIMEOUT 5 ABORT ‘ERROR’ ‘’ ‘’ OK-+++\c-OK ATH0 ‘’ ‘AT’ OK ‘ATD#777’ TIMEOUT 180 ABORT ‘NO ANSWER’ ABORT ‘NO CARRIER’ ABORT ‘NO DIALTONE’ ABORT ‘BUSY’ CONNECT ‘\c’	DEVICE = /dev/ttyS0 DEVICE_SPEED = 115200 MODEM_INIT = AT PHONE_NUMBER = CONNECT_TIMEOUT = 180 USERNAME = PASSWORD = ALWAYS_UP = no DEFAULT_ROUTE = yes ENABLED = yes SHELL_LOGIN = no SEND_CR = no

/etc/config/diald.conf
mode ppp connect “/bin/chat –v –f /etc/config/chat.dialout” device /dev/ttyS0 speed 115200 modem lock crtscts local 192.168.1.1 remote 192.168.1.2 dynamic defaultroute include /etc/config/diald.dynamic

Save the changes to flash memory.

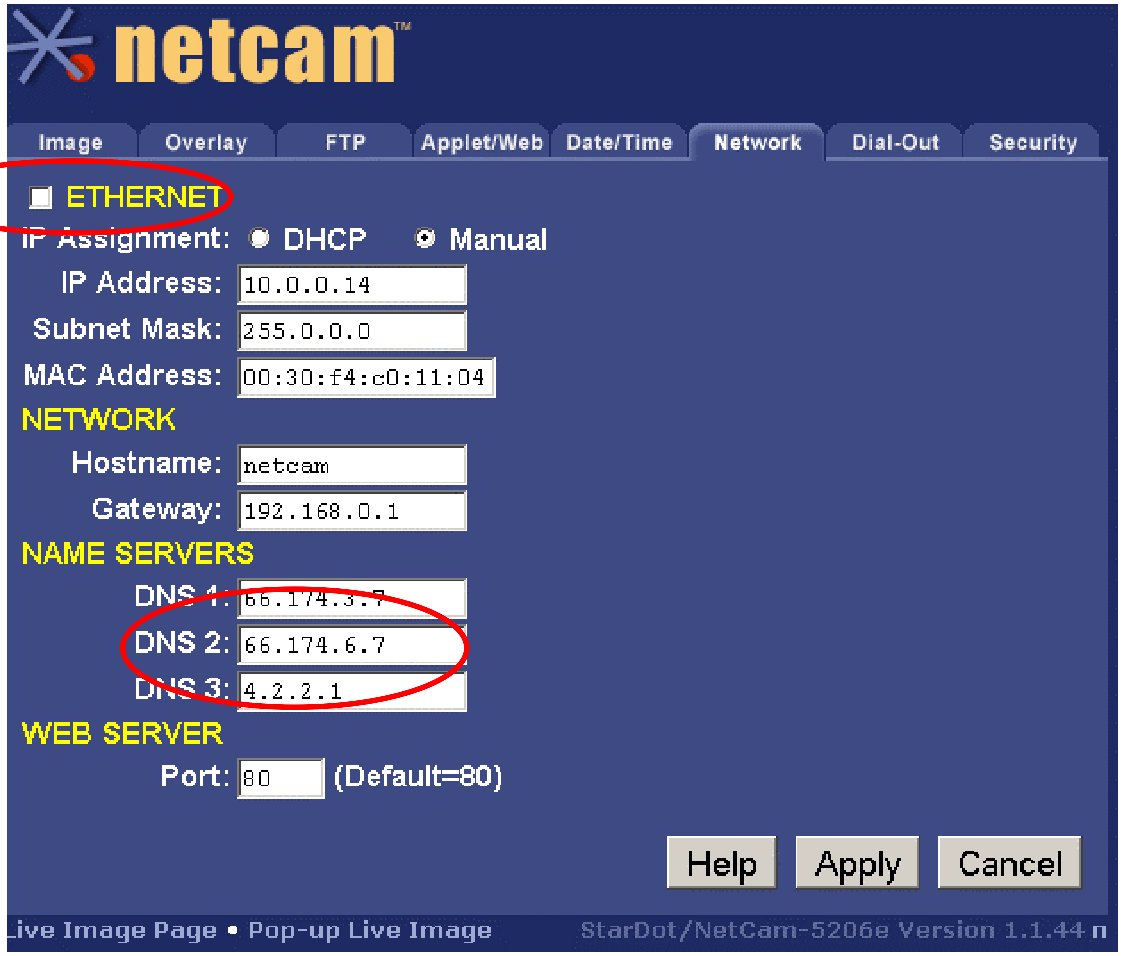
Enabling FTP Upload

According to Stardot Corp., FTP upload is required during modem operation. Go back to the FTP configuration screen and enable an FTP upload to the main FTP server. It is helpful to include wild cards in the filename to assist with time and date coding. For example, use “%Y” to list the four-digit year or “%H” to list the two-digit hour. The StarDot Communications Netcam manual provides additional information about these features.

Last Configuration Changes for Network Access

The **VERY** last configuration settings to be changed are the Network settings. The Ethernet port must be completely disabled in order for successful dial out modem capability. This is because the NetCam will automatically use the Ethernet port for connectivity if it is enabled on the configuration web page. The NetCam favors the Ethernet port for its default Internet route whenever there are multiple connections to the Internet. Also, remember to enter the correct DNS servers for the ISP being used. Although the

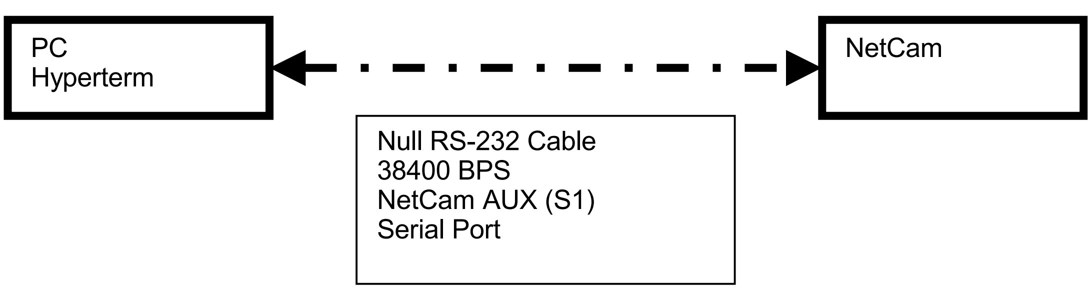
modem will obtain IP information from the ISP, the NetCam's firmware prevents it from dynamically assigning the DNS servers. Instead, the NetCam uses a static file "/etc/config/resolv.conf."



Once these settings are applied, communications with the NetCam via the configuration web page will be impossible. Camera must be rebooted to observe if changes are successful.

Diagnostic Aids

It is helpful to have the NetCam connected to a PC with an RS-232 null cable to diagnose any boot problems. Most specifically, check the operational status of PPPD (the peer to peer protocol daemon).



Connect the null RS-232 cable to the NetCam's auxiliary serial port (S1). The other S0 serial port should already be connected to the modem. Log into the NetCam with a serial communication client, such as HyperTerm ™, and set the direct serial connection settings for 8N1 Hardware flow control at 38400 BPS. Once a connection is initialized, hit <ENTER> to receive a NetCam login prompt. Check the cable and baud rate settings if a prompt is not received. Otherwise, log in to the NetCam with its username and password (Addendum A).

After logged into the NetCam, view the processes running with "**ps –auwx**". Use the Linux command "**ifconfig**" to find out interface information. If the NetCam is correctly configured, this command will list information about the PPP connection. If a PPP connection exists with the correct IP address, then the Internet connection through the modem should work. Try to use "**ping**" to reach a network computer or node. If the "**ping**" command has a timeout, it may be due to incorrect DNS settings (if so, modify the /etc/config/resolv.conf nameservers) or incorrect modem initialization settings.

Temporary NetCam Access through Ethernet Port

It is possible to communicate with the NetCam through the network interface temporarily without erasing the existing settings. This is necessary when positioning the camera with a local laptop. Issue the follow commands from a serial port connection:

1[~]# *killall pppd*
2[~]# *killall diald*
3[~]# *ifconfig eth0* **10.0.0.14** *netmask* **255.0.0.0** *up*
4[~]# *route add default eth0*

The first line stops the pppd daemon from running. The second line stops the diald daemon from running. Stopping these daemons makes it easier to configure the network interface as the temporary default route. The third line brings up the Ethernet interface with IP address **10.0.0.4**. Change the IP address to one that is routable on the network and specify the corresponding network mask. The last line tells the operating system to use the Ethernet port as the default route. This is required to properly route the packets through the NetCam's Ethernet port.

Now there is access to the NetCam's web page through the Ethernet port at the IP address specified until the camera is rebooted. The camera will load its normal configuration settings from flash memory on the next boot. The NetCam's permanent settings regarding network and dial-up access through its own web pages can be adjusted via this interface is desired. For further information on camera configuration methods and manufacturer updates please contact http://stardottech.com/netcam/index.html.

5.4.6 Visibility Sensors

The visibility sensor data is sent along with other weather station data over a RS232 serial communications line. A standard 4-conductor telephone cable is used to transfer the RS232 signals from the weather station into the 222 Cabinet. Only the output data line from the weather station is connected to the RS232 input port of the evaluation computer. This prevents the evaluation computer from interfering with the data communications from the weather station to the QCMS computer in the TMC.

A sample of the RS232 data packet stream is as follows:

```
62000000000000000181 23:55:09 0.23 0440 0005 --.-- -- ---- 8 4841 11718  27 -+--5849
```

The weather station generates a new visibility reading every 30 seconds, but it sends out that same reading multiple times over a 30 second period. The time value (bytes 25 – 32) is extracted from each packet and used as a key to detect when a new packet arrives. When a change in the time value is detected, the visibility coefficient (bytes 34 – 39) is extracted from the RS232 data stream, time-stamped, and stored in the evaluation computer database for future transfer over to the central server.

Further information on the packet data is available in the 8364 Qualimetrics Visibility Sensor manual.

A limited weather station is deployed at the French Camp Slough site. Only the visibility sensor and day/night sensor are installed. However, the full weather station is deployed at the Mathews Road site because it is part of the CAWS. The full weather station includes additional sensors, including an anemometer, barometer, and temperature sensor.

The laptop diagnostic port may interfere with the auxiliary port connector in the QNET electronic box.

5.4.7 Day/Night Sensors

The All-weather Systems weather stations include a binary-output illumination level sensor referred to as a day/night sensor. Its input is used by the weather system to apply the appropriate visibility formula relating the extinction coefficient to FAA-type visibility reported in feet or meters. We interface these sensors directly at the French Camp Slough and Mathews Road sites via a custom-designed general-purpose loop detector interface. The device connects to a computer's parallel port and requires straight wire connections to the screw terminals of a loop device (where the IST 222 cards plug in). The interface module also contains an input from the day/night sensor, to report the illumination condition as required in the CAWS evaluation database.

Normally, a computer’s parallel port is used for printing only (not data input). If the parallel port is set for printing use, then the data lines on the parallel port will be used for the output of data only; the parallel port is in “SPP” mode. However, it is necessary to set the parallel port data lines to bidirectional for data acquisition. This is done in the computer's BIOS by setting the parallel port option to “ECP+EPP”.

	Wire Color
Day/Night Data	Purple
GROUND	Black

The Day/Night sensor provides a TTL logic level signal for easy interface through the general-purpose loop interface. The TTL level transition from logic 1 to 0 indicates that the sensor has detected daybreak. The TTL level transition from logic 0 to 1 indicates that the sensor has detected evening.

5.4.8 Inductive Loops

The loop detector cards are plugged into the loop detector backplane. This backplane routes +24VDC power to the loop detector cards and allows the 170 controller to have access to the signals. Loragen has

modified the power distribution to the loop backplane to supply UPS battery backup power to the loop detectors. A +24VDC table-top 1 AMP AC adapter replaces the +24VDC power source supplied from the 170 Power Distribution Assembly. The table-top adapter is plugged directly into the UPS.

Outside of the scope of the CAWS evaluation contract, Loragen has developed a general-purpose parallel port data interface module, which allows real-time data to be acquired and recorded in a time-accurate manner using the parallel port of a PC running Linux. We use this device to interface the 222 loop detectors, and perform all timing measurements required for precise vehicle time-of-arrival and speed measurements. The device driver that communicates with the interface hardware runs on Linux platforms only. The loop interface routes the loop detector signals (leading and trailing lines) to the parallel port's data lines. The loop interface generates a parallel port interrupt on any loop signal transition. This allows the device driver to check the parallel port lines and send vehicle information to a reading process.

		Wire Color
1 [FAST]	Lead	Brown
		Red
2	Lead	Orange
		Yellow
3 [SLOW]	Lead	Green
		Blue
	GROUND	Black

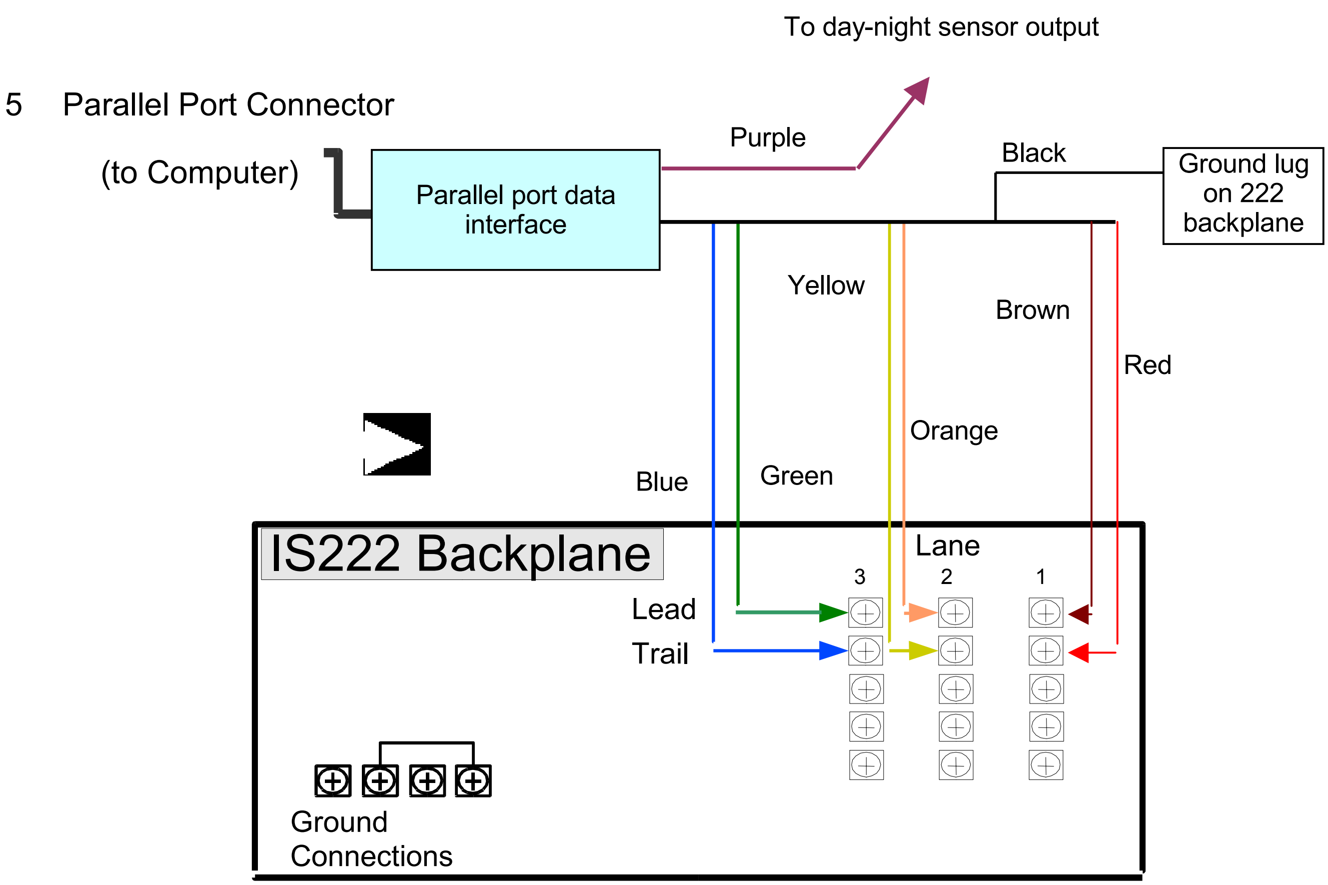


Figure 5.4.8.1. Interface of loop detectors to PC DAS via parallel port adaptor.

The parallel port loop interface is installed by connecting the wires from the device to the screw terminals on the back of the loop interface rack-mount unit. The wires on the loop interface are color coded and labeled to help ensure installation to the correct lanes.

Newer revisions (all currently installed) of the parallel port loop interface device contain external connections for the power and ground signals. The longer, red power wire should be connected to a +24VDC power source. Previously, the parallel port pin on the computer motherboard was modified to send +5V from the computer's power supply to parallel port pin #13 S4 (Printer Selected).
Figure 5.4.8.2 shows the installation of the parallel port loop detector interface installed in a typical field cabinet.

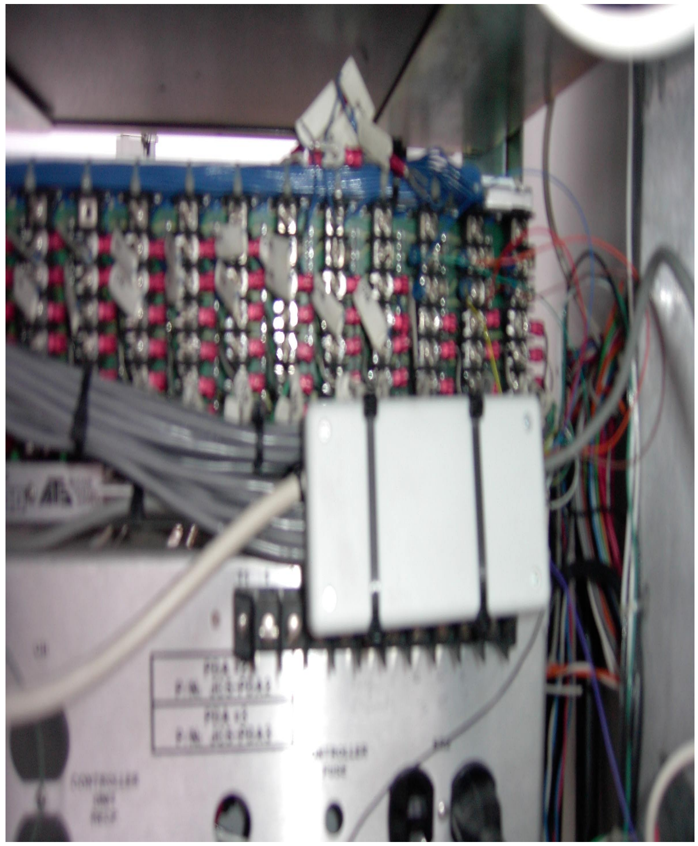


Figure 5.4.8.2. Parallel port interface adaptor for loop detectors, installed on 222 detector backplane.

5.5 Data Acquisition System Software

The software components are made up of the system software and the Loragen modules.

5.5.1 System Software

The system has the boot loader GRUB (grand unified boot loader) installed. Previous iterations of the system software used LILO (Linux loader), which proved to be inflexible in many cases. LILO requires direct modification of the boot sector on the boot partition whenever a change is made to the boot process. GRUB, however, allows instant boot changes through a single configuration file **grub.conf**. GRUB loads boot information directly from the configuration file when the system boots.

```
title Red Hat Linux *D10 APM flash/ramdisk * root=/dev/ram3 (2.4.18-10)
   root (hd0,0)
        kernel  /vmlinuz-2.4.18-10-APM vga=791 root=/dev/ram3 ramdisk=786432
init=/linuxrc ro
   initrd /rdimg11-flash.gz
```

This boot script entry in the grub configuration file loads the boot information for the data acquisition system. Advanced Power Management (APM) is required for automatic power-off at shutdown (used in the UPS Monitoring Module (`lgups`)). The system uses the ramdisk /dev/ram3 as its root partition after it completes the boot process. The Linux kernel version 2.4.18-10 is used with APM enabled. The boot option “vga=791” allows a VGA text console to be used at 1024x768. It is extremely helpful to see more than 80 lines of console text at a time. The system will execute the “linuxrc” file after it uncompresses the small initial boot ramdisk “rdimg11-flash.gz”. The “linuxrc” file is used to uncompress and copy the contents of the large system image into the 768MB ramdisk.

All the system files necessary for boot are located in the first partition of the flash disk (/dev/hda1). This is where both the initial (small) boot ramdisk and (large) compressed system image reside. The boot sector of the flash disk has been preloaded with the GRUB boot loader, as discussed previously. Although unused at this time, a second partition exists on the flash disk (/dev/hda2) that could be used to store logging information.

A disk partition table is shown in Figure 5.5.1.1. The hard disk has an 18 GB partition that is used for storage of data when an Internet connection fails. Currently, the system log file is located on that partition as well. In addition, a 2 GB Linux operating system is installed on the second hard disk partition to facilitate making on-site repairs to the boot process. The second hard disk partition will not be used in normal operation.

After the Linux kernel is loaded into memory, the initial (small) boot ramdisk is uncompressed and loaded into ram. On normal systems, this small initial ramdisk is used to load optional kernel modules for the system, such as SCSI disk drivers that the kernel uses to find a root disk drive. Loragen specially modified this initial (small) boot ramdisk for the data acquisition systems. The Linux kernel immediately looks for a script file in the uncompressed contents of the small boot ramdisk. This script file is called **linuxrc**.

The initial (small) boot ramdisk **linuxrc** script has been modified to check the flash and hard disk partitions with the *e2fsck* (extended second file system checker) Linux utility. It automatically checks the flags that the *e2fsck* utility returns to the system so that an automatic reboot can be accomplished immediately if required. This was implemented so that the computer would not freeze during boot or during the rare case where the file system checker corrected errors on the partitions.

The **linuxrc** script has also been modified to locate and uncompress the large system image file (stored on the flash IDE drive) to a ramdisk. Then, the script finds all the modification TAR files in the /boot/mods/ directory of the flash IDE drive and extracts them on top of the recently created system image in RAM in alphabetic order. Then, the script completes and the Linux kernel deallocates the memory used by the initial (small) boot ramdisk and continues system initialization from the ramdisk (/dev/ram3). From this point on, the system no longer needs any information on the flash drive to operate. The system continues to run from RAM and acquire data with its data collection programs.

The system will be booted very few times because the system is intended to remain on for extended periods. This system must boot or reboot when:

1). System boot parameters are changed or the kernel is reconfigured

2). A lengthy (3+ hours) power outage occurs and the system shuts down (by turning the UPS off). When power is restored, the UPS power is restored and the system BIOS has been configured to automatically allow the computer to boot.

3). The system hangs due to a hardware malfunction or problem with the Linux operating system kernel. In this case, the watchdog timer card will reboot the computer.

After the system boots, it mounts the hard disk drive (/dev/hdc1) to the **/backup** mount location. If the hard disk drive or IDE interface malfunctions, then the automatic backup script (`lgbkp`) may experience problems backing up data to the hard disk partition. In this case, the backup module will abort the backup of data to the hard disk and the data will remain in ram.

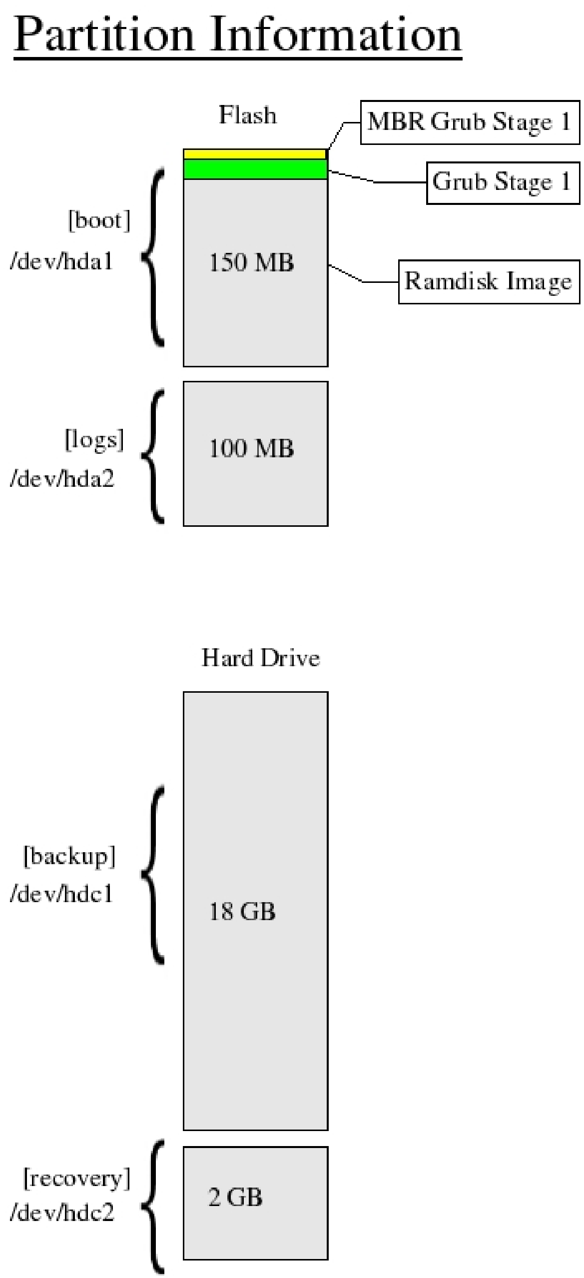


Figure 5.5.1.1. Hard disk Linux partition information, DAS computers.

5.5.2 Configuration File (`loragen.conf`)

Location

`/etc/loragen.conf`

Description

The `loragen.conf` configuration file is used to set the defaults to be applied when running the Loragen modules.

Options

The different configuration options are:

`ID=<site-id>`

Used to specify the site ID of the field unit. Table 5.5.1 shows the site IDs and their corresponding names.

Table 5.5.1. Site IDs and Names.

	Site Name
1	French Camp Slough
2	Mathews Road
3	French Camp CMS
4	Downing Road
5	El Dorado Overcrossing
6	Spare Unit

`SIGN=<sign-number>`

Used to specify the CMS number that will be processed by the sign data collection module (see SS 5.5.15).

5.5.3 Module Start/Stop Script (`loragen`)

Location

`/etc/init.d/loragen`

Usage

`loragen {start|stop}`

Description

This script is used for automatically starting and stopping the Loragen modules. If installed with `chkconfig` (`/sbin/chkconfig --add loragen`), it will be started in run levels 3 and 5, with a start priority of 95 and a stop priority of 5. Though the modules can be started and stopped individually, it is recommended that this script be used to ensure that multiple instances of a module are not run concurrently, which could result in resource conflicts.

When this script is invoked with "`start`", it performs the following:

1. Reads the configuration information from `loragen.conf` (see SS 5.5.2).
2. Creates the module log file (`/home/loragen/loragen.log`), if it doesn't exist.
3. Runs `setMyIdentity` (see SS 5.5.7) to set the identity of the field unit.
4. Starts `setMyIP` (see SS 5.5.8) to post the system IP address to the central server.
5. Enables the watchdog timer card in the Linux kernel.
6. Starts the UPS monitor.
7. Loads the loop interface device driver.
8. Starts the backup and data acquisition modules in the background.

When invoked with "`stop`", it performs the following:

1. Stops the backup and data acquisition modules.
2. Unloads the loop interface device driver.
3. Disables the watchdog timer card in the Linux kernel.

Some limitations on the use of this script are:

1. It must be run as the superuser.
2. It should not be started until after the MySQL server is active. If the MySQL start/stop script has a lower start priority than this script, the modules will not start correctly.
3. It will not stop `setMyIP`. This step was taken to avoid losing contact with the field units with analog modem connections and dynamic IP addresses.
4. It will not stop the UPS daemon lgups. The system should still power down safely if it goes on battery backup power and the low-battery signal is received.

5.5.4 Modem Connection Script (`beOnline`)

Location

`/usr/local/bin/beOnline`

Usage

`This script is invoked when networking is brought up (/etc/init.d/network).`

Description

This script supplies modem hardware configuration options (such as COM port and baud rate settings) to the PPP connection daemon. This script initializes a connection to a CDPD modem or a standard telephone modem. The script continuously tries to connect to the Internet. If the Internet connection disconnects, the script will try to persistently attempt to reconnect until a new connection is made. This script calls the **watchPPP** (watchdog) script to aid in determining if the Internet connection is stable. The **beOnline** script is machine specific based upon Internet connection availability. In the event that a configurable number of attempts to connect via the wireless CDPD modem have failed, the script attempt to establish an alternative connection via the POTS telephone modem for the sites at which telephone lines are present (CMS and Mathews Road).

5.5.5 Connect Script (`connect`) and Chat Script (`chatscript`)

Location

`/usr/local/bin/connect`

`/etc/ppp/chatscript`

Usage

```
The connect script is invoked automatically by the PPPD daemon.  The connect script
calls the chatscript when a connection has been established.
```

Description

The chat script contains the Internet connection specific communications. If a dial-up Internet connection is used, this file will contain the user name and password information to authenticate the system with the ISP. If a CDPD modem connection is used, the modem compatible PPP link initialization parameters are used. Hayes standard AT commands are used in this file to interface with either modem type.

5.5.6 Network Interface Configuration (`ifcfg-eth0`)

Location

```
/etc/sysconfig/network-scripts/ifcfg-eth0
```

Usage

```
This file is used by the network daemon.
```

Description

This file specifies the IP address and network information for the Ethernet port on the computer. Each computer has its own unique IP address on the private Class A network (IP Address 10.0.0.X with netmask 255.0.0.0). Table 5.5.2 shows the private Ethernet IP addresses for the field sites.

Table 5.5.2. Ethernet IP Addresses.

		Ethernet IP Address
1	French Camp Slough	10.0.0.1
2	Mathews Road	10.0.0.2
3	French Camp CMS	10.0.0.3
4	Downing Road	10.0.0.4
5	El Dorado Overcrossing	10.0.0.5
6	Spare Unit	10.0.0.6

5.5.7 Identity Setting Script (`setMyIdentity`)

Location

```
/usr/local/bin/setMyIdentity
```

Usage

This script is invoked from the **loragen** boot service (/etc/init.d/loragen)

Description

This script sets the hostname and inserts the configuration information into the database. It inserts the ID number of the machine into the database. The machine ID number is used specifically in the *IP Setting Script.*

5.5.8 IP Setting Script (`setMyIP`)

Location

`/usr/local/bin/setMyIP`

Usage

`This script is invoked from the` **`loragen`** `boot service (/etc/init.d/loragen).`

Description

This script periodically reports the field unit's IP address to the central server. The script runs on the field machine and calls the "**set_ip.php**" webpage script on the central server to report its IP address and ID number.

5.5.9 Network Monitoring Script (`watchPPP`)

Location

`/usr/local/bin/watchPPP`

Usage

`This script is invoked from the` **`beOnline`** `script (/usr/local/bin/beOnline).`

Description

This script monitors the network connection, and if down, requests that a new modem connection be made. This script monitors the **beOnline** script's PPP session. If the connection times out (meaning the script is unable to ping the central server and other well known locations) the connection will be terminated and a new connection will be started.

5.5.10 Automatic Backup Script (`lgbkp`)

Backup Module (`lgbkp`)

Location

`/usr/local/bin/lgbkp`

Usage

```
lgbkp [-i=BKP_INTERVAL] [-a=BKP_AGE] [-dh]
   -i --interval=BKP_INTERVAL  set how often this module checks for data to backup to
                                BKP_INTERVAL seconds [default: 60]
   -a --age=BKP_AGE             set the minimum age of data to backup to BKP_AGE
                                minutes [default: 5]
   -d --debug                   print debug output
   -h --help                    print this message, and exit
```

Description

Under normal system conditions, each iteration of the `gather` script (see Section 0) will collect all the available data from the `field2` database, leaving the tables empty. Network outages or problems with the central server could prevent the script from executing, in which case all newly acquired data will accumulate on the field unit's ramdisk. When the gather connection is restored, it will eventually return the unit to its normal state. But if, for some reason, the data remains ungathered for a prolonged period of time, the ramdisk could fill up and stop storing new data. More importantly, if the system were to crash, all the volatile ramdisk data would be lost. The backup module attempts to resolve these issues.

`lgbkp` periodically moves "aged" data from the database on the ramdisk (`field2`) to the backup database on the large, non-volatile hard drive (`fieldbkp`). Data is considered "aged" if it has not been gathered in `BKP_AGE` minutes (set on the command line, or 5 by default). With this approach, there is no risk of filling up (the hard drive partition containing `fieldbkp` is 20 times larger than the ramdisk), and in the event of a system crash, no more than `BKP_AGE` minutes of data are lost.

Every `BKP_INTERVAL` seconds (set on the command line, or 60 by default), `lgbkp` attempts to back up the seven data tables (`day_or_night_data`, `fog_data`, `image_data`, `sign_data`, `site_status`, `speed_data`, `watchdog_data`). For each table, the steps are as follows:

1. Write-lock the table in `field2` and `fieldbkp` to eliminate race conditions.
2. From the `fieldbkp` table, select the most recently inserted timestamp. This is done to avoid backing up duplicate data.
3. From the `field2` table, select all "aged" data younger than the timestamp from step 2.
4. If no data is returned, skip to step 7. Otherwise, continue to the next step.
5. Insert the data into the `fieldbkp` table.
6. Delete the data from the `field2` table.
7. Unlock both tables.

This approach solves the following problems encountered in the first implementation of the backup module, which periodically copied the *entire* database to a backup partition and restored it on system initialization.

1. In the event of a network outage, the backup module would take increasing amounts of time to copy the database files, due to the increasing number of records in the tables. The locks held on the tables during that time could cause the data acquisition modules to block on an insert, and potentially miss data. The current approach only copies a small amount of data, minimizing the lock time.
2. The database size was limited to the size of the ramdisk. If the ramdisk filled up and caused a system crash, the full database would be restored when the system was reset, potentially causing another crash. With the current approach, the database can grow to fill the larger hard drive partition, and there is no need to ever copy the data back to the ramdisk.
3. If the copy failed, or was somehow corrupted, the previous database backup would be overwritten with garbage. If the system crashed before this situation was rectified, all data collected during

the network outage would be lost. The current approach never overwrites previously backed up data, avoiding this situation entirely.

The current implementation of `lgbkp` is not without its share of problems, though none are serious enough to raise any concern.

1. The tables `lgbkp` can back up are limited to those with the following structure:
   - The table column order in both the `field2` and `fieldbkp` databases **must** be the same, as the insert commands do not specify column names.
   - The first column of the table **must** be the ID, an automatically incrementing primary key. It will be set to `NULL` when inserted into the `fieldbkp` table to avoid having duplicate IDs.
   - The table **must** have a timestamp column called "stamp".

2. Since `lgbkp` only backs up and deletes "aged" data, it will never delete *all* the records from a `field2` table. Therefore, the only way the automatically incrementing IDs in the `field2` and `fieldbkp` tables will be reset is if all the records have been gathered from them. If the network is down or the gather routine is inactive, the IDs will continue to increase until reaching their maximum values, after which all new entries will be discarded. Table 5.5.3 details the theoretical lengths of time before this condition is reached.

Table 5.5.3. Estimated Times Before Reaching Maximum IDs.

			Est. Time
`day_or_night_data`	`tinyint unsigned`	2/day	127 days
`site_status`	`smallint unsigned`	12/hour	227 days
`parallel_speed_data`	`mediumint unsigned`	3000/hour	233 days
`fog_data`	`mediumint unsigned`	2/minute	5825 days
`sign_data`	`smallint unsigned`	10/day	6553 days
`image_data`	`mediumint unsigned`	72/hour	9709 days
`watchdog_data`	`smallint unsigned`	6/hour	455 days

3. In the unlikely event that `lgbkp` backs up data and fails to delete it from the `field2` table, duplicate data may be gathered from the field unit. This will only occur if the gather connection is restored before the next data backup and subsequent successful deletion of the duplicate records. If this were to happen, the situation could be rectified through the use of a script on the central server.

5.5.11 Loop Data Interface Driver (parallel_loop_driver)

Much of the development on the device driver for the parallel port loop interface came from a well-known book, Linux Device Drivers by Alessandro Rubini and Jonathan Corbet (published by Oreilly). Most of this book is now available for free on the Internet.

The code examples provided in the book are supplied under the GNU public license. Since most of the concepts were taken from sample code provided in the book, the device driver itself should be considered under this license.

The device driver has been compiled to run on version 2.4.18-10 of the Linux kernel. A custom kernel configuration file has been developed for this kernel that eliminates the need for extraneous kernel modules. Therefore, the only kernel module needed is the one for this particular device driver.

The device driver installs an interrupt service routine (ISR), or handler, for the parallel port. Loragen has customized the driver code to provide time stamped event information for each data line signal transition

on the parallel port. The parallel port offers 8 data lines D0-D7. Currently, the device driver expects two loop signals for each of the three lanes (for a total of 6 signals). D0 and D1 correspond to the leading and trailing loop lines for lane #1 (the fast lane). D2 and D3 correspond to the leading and trailing loop lines for lane #2 (the middle lane). Similarly, D4 and D5 correspond to the leading and trailing loop lines for lane #3 (the slow lane).

A *STRUCT carInfo* or 'C' structure of information is used to represent each car that travels down the freeway. Information in the structure is as follows:

startTime: The time the car first appears at the first (leading) loop.

endTime: The time the car first appears at the second (trailing) loop.

classification1: The time the car leaves the first (leading) loop.

classification2: The time the car leaves the second (trailing) loop.

lane: The lane number of the vehicle (1=fast, 2=middle, 3=slow)

placeHolders: This values are included in the structure to force the structure into a size that's a power of two. This is required for kernel memory paging operations. This is required for proper operation of the circular buffer routine inside the driver.

Each *TIME STRUCT* contains the Unix equivalent timestamp information. Two fields are provided in the *TIME STRUCT.* The "Seconds" field represents the total number of seconds since January 1, 1970. The "Microseconds" field represents the total number of microseconds after the current number of seconds. This information can be used to generate a near microsecond resolution timestamp.

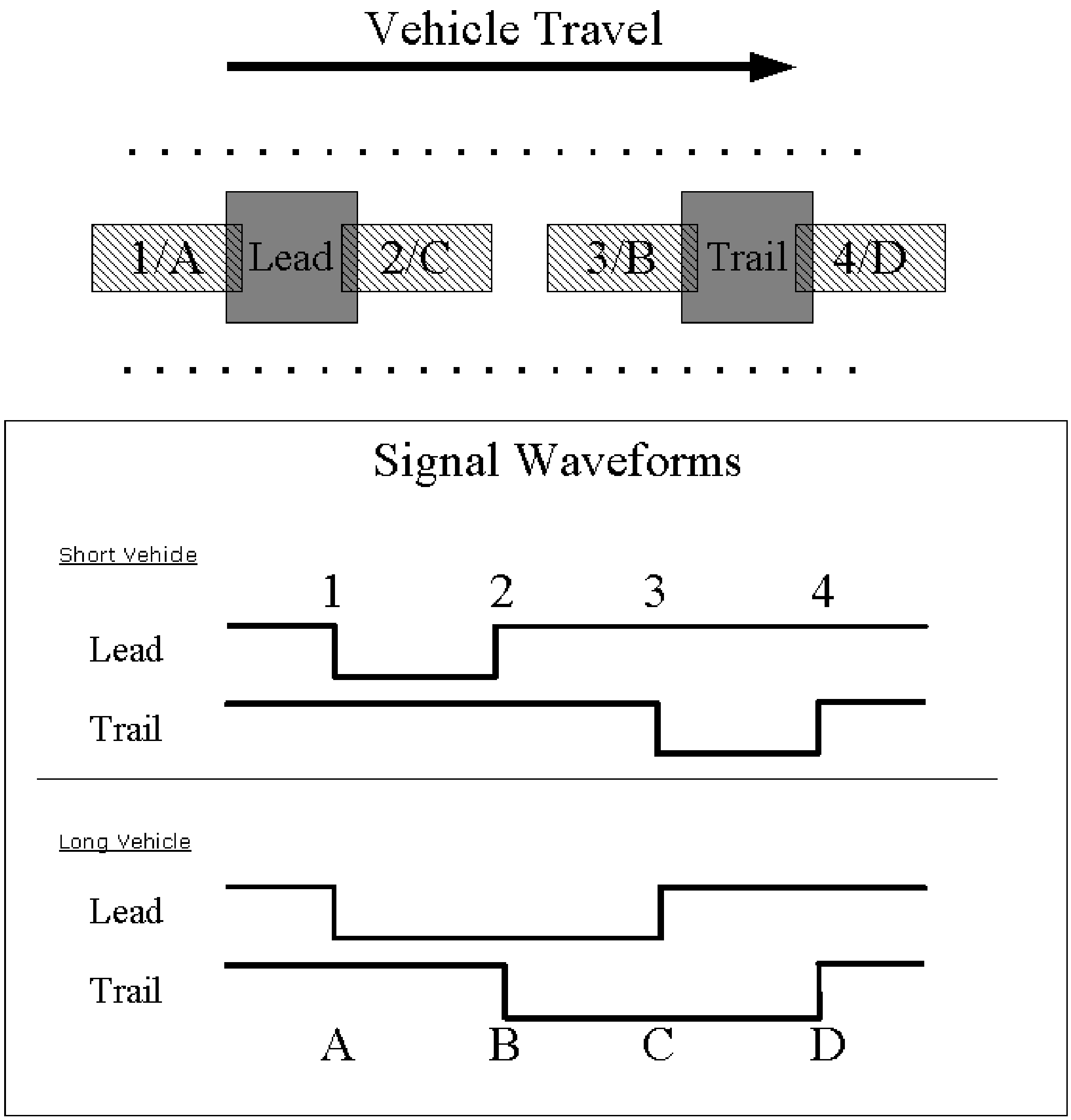


Figure 5.5.11.1. Loop signal events.

The hardware loop interface device generates an interrupt when a signal transition is detected on the data lines. At this point, the CPU starts to execute the custom interrupt handler for the parallel port. The interrupt handler timestamps each signal transition on the data lines. All the necessary data line transitions have occurred after a vehicle leaves the second (or trailing) loop. This is indicated by the rising edge of the trailing loop data line. During this particular transition, interrupt handler dispatches the vehicle data structure into a circular queue to be read by the client data collector program.

The first signal transition occurs on the falling edge of the leading loop data line. This event is time-stamped to indicate the time that the vehicle has first appeared at the loop. The second and third signal transitions will indicate the moment the vehicle arrives at the trailing loop or the moment that the vehicle has finished crossing the leading loop. Longer vehicles will reach the trailing loop before completely crossing the lead loop. Shorter vehicles will finish crossing the lead loop before reaching the trailing loop.

The last signal transition occurs on the rising edge of the trailing loop data line. This will always indicate that a vehicle has completely passed the speed site.

Since the custom parallel port driver only generates event times, it is necessary for the client program to subtract the event times to determine time lengths. The timestamps in the vehicle data structure are subtracted appropriately in the loop data collector client program. The client program subtracts the timestamps to generate the vehicle travel time between the loops and the two classification times (the travel times across the individual loops).

The Linux device driver is compiled as a kernel module. This means that the driver can be loaded into memory or unloaded from memory at any time while the system is running. This enables easier development of the device driver because the device driver does not have to be recompiled into the kernel after each alteration. Two scripts were adapted from Linux Device Drivers that automate the loading of the device driver.

load_loop

Automated script that loads the Linux parallel port loop interface device driver.

unload_loop

Automated script that unloads the Linux parallel port loop interface device driver.

The Linux utility “lsmod” can be used to determine if the kernel driver module is currently loaded.

5.5.12 Fog Data Collection Module (`lgdcf`)

Location

`/usr/local/bin/lgdcf`

Usage

```
lgdcf [-t=SERIAL_TIMEOUT] [-dh]
   -t --timeout=SERIAL_TIMEOUT   set the timeout for reading serial data to
                                 SERIAL_TIMEOUT seconds [default: 300]
   -d --debug                    print debug output
   -h --help print this message, and exit
```

Description

This module reads fog data from a serial port and stores it in the `fog_data` table in the `field2` database.

The packets generated by the Qualimetrics 8364 visibility sensor are 103 bytes long, begin with three `SYN` (`0x16`) bytes, and end with `CR` (`0x0D`), `LF` (`0x0A`). The data segment is 98 bytes long, and is in the following format:

```
62000000000000000181 23:55:09   0.23 0440 0005 --.-- -- ----      8   4841  11718     27 -+--
5849
```

The only fields used by this module (the full packet breakdown can be found in the 8364 visibility sensor documentation) are the time (bytes 25 - 32) and the extinction coefficient in inverse-miles (bytes 34 - 39). The time is not consistent with the synchronized system time on the field units, and is used only to distinguish between packets. The visibility sensor generates a new fog data packet every 30 seconds, and then serially transmits it multiple times until the next packet is generated. To avoid inserting redundant entries into the table, this module will only insert a coefficient if its time is different than that of the last inserted coefficient.

On initialization, `lgdcf` does the following:

1. Makes a permanent connection to the `field2` database.

2. Gets the `COMPONENT_FOG` configuration information from the `config` table. If the information does not exist, `lgdcf` will exit. The `source` value should be set to the serial device that is connected to the visibility sensor. The `delay` value is not used.

3. Makes a 1200-baud, 8-N-1 connection to the serial device determined in step 2.

The fog packets are processed as follows:

1. `lgdcf` calls `select()` to wait for data on the serial connection. If `SERIAL_TIMEOUT` seconds (set on the command line, or 300 by default) pass without receiving data, an error message is printed and this step repeats.

2. The received data is read into a 255-byte character buffer. If the total number of bytes in the buffer is greater than 103 (the size of a valid fog packet), it proceeds to the next step. Otherwise it returns to step 1.

3. The data in the buffer is analyzed for the correct packet header and footer. If found, it proceeds to the next step. Otherwise it returns to step 1.

4. The time and extinction coefficient are extracted from the packet.

5. The time is compared to the time of the last inserted coefficient. If they are the same, it proceeds to step 8. Otherwise it continues to the next step.

6. The coefficient is inserted into the `fog_data` table. If the coefficient is less than 0.15 mi^{-1}, it will be transmitted as `<.15`. If this is the case, it will be converted to `0.15` before inserting it. The timestamp will be set automatically by MySQL.

7. The time is saved as the time of the last inserted coefficient.

8. The buffer is emptied and the process is repeated.

When lgdcf is stopped with a SIGINT or SIGTERM signal, it will cleanly exit, closing the serial device and the MySQL connection.

The following design decisions were made when writing this module:

1. It uses `select()` instead of interrupt-based I/O. It was found that if an interrupt occurred while inserting data into MySQL, the insert would fail. With `select()`, there is no such problem.

2. No input processing (non-canonical) is performed on the serial data. All received bytes are stored in a buffer, which will be processed upon reaching a reasonable size. With this approach, if there are two

packets in the buffer at the time of processing, one will be lost. This is not an issue as each packet is transmitted multiple times. An earlier version of `lgdcf` that used canonical input processing would not reliably read full fog packets, resulting in data corruption.

5.5.13 Image Collection Module (`lgic`)

Location

`/usr/local/bin/lgic`

Usage

`lgdcf`

Description

This program collects images from web cameras and loads them into the MySQL database.

5.5.14 Watchdog Data Collection Module (`lgic`)

Location

`/usr/local/bin/lgwdt`

Usage

`lgwdt`

Description

This program collects information about boot status and system temperature and stores it into the MySQL database. In addition, this program accesses the watchdog timer kernel driver information to periodically reset the watchdog timer. Access the **/proc** kernel interface for this device driver resets the watchdog timer. If the watchdog timer card has not been read from in XXXX, it will automatically reset the system.

Data type field in the database can contain either two types of data. 'B' data means boot data: the watchdog timer card stores information about whether or not it caused the last reboot of the system. 'T' data refers to temperature data. The watchdog timer card supplies a temperature sensor and redundant temperature information is stored.

The watchdog data collection module is currently set to gather temperature data every 10 minutes. The watchdog data collection module retrieves the time period for temperature sensor measurements from the configuration table in the MySQL.

5.5.15 Sign Data Collection Module (`lgdcs`)

Location

`/usr/local/bin/lgdcs`

Usage

`lgdcs`

Description

This program interfaces with the RS-232 serial port to monitor messages sent to the County Hospital CMS (CMS 1).

5.5.16 Status Monitoring Module (`lgm`)

Location

```
/usr/local/bin/lgm
```

Usage

```
lgm [-dh]
   -d –debug                     print debug output
   -h –help  print this message, and exit
```

Description

The monitoring module runs operating system calls to determine information about the computer's status.

5.5.17 UPS Monitoring Module (`lgups`)

Location

```
/usr/local/bin/lgups
```

Usage

```
lgups [-dh]
   -d –debug                     print debug output
   -h –help  print this message, and exit
```

Description

This program constantly monitors the UPS to detect when the system is on battery and when the battery is low. Once the emergency condition is reached, this program will execute a "**telinit**" system call to place the computer in a shutdown state.

The shutdown state modified is Linux Run Level #4. Typically, this run level state is unused by the system. Loragen has modified this run level state so that it is identical to the run level used by **HALT**. However, the last process executed in the shutdown state is **lg-ups-critical-binary**. This utility is run to prevent race conditions from occurring such as attempting to shut the UPS down after power has been restored.

The UPS cannot be turned off when power is supplied to it. The UPS can only be shutoff when it is on battery backup power. Therefore, any attempt to shutdown the UPS by using an RS-232 signaling pin will result in failure if the power has already been restored. If power has been restored, the computer will perform a reboot (similar to <CTRL><ALT><DELETE>) in order to restart its processes. Otherwise, the computer will attempt to shutoff the UPS by signaling a logic low level on the shutdown pin. If the computer is ever unsuccessful at shutting itself off via the UPS (it is still executing the program) it will reboot.

The computer does not shut itself off directly. If it did, there would be a possibility that the computer won't be automatically powered back on when power is restored. This would only happen if the UPS did not automatically turn itself off after a period of time, but we avoid this situation completely with our solution.

Note: The motherboards have the capability to automatically turn on the system when power is restored. We rely on this capability when we turn off the ups because of a low battery condition. The UPS will automatically turn back on and resume charging when power is restored to a site.

5.6 Central Server

The CAWS Evaluation Central Server is physically a PC-class system with an AMD ATHLON XP 1800+ processor running Red Hat Linux. It is designed for maximum data reliability, incorporating dual SCSI hard drives which mirrored – the same image is maintained at all times on both. In the event that one drive fails, no data is lost and the system continues to operate. A 400 Watt high-reliability power supply is used. The system also includes a CD writer (CDRW) for data backup to CD.

The Central Server runs the main evaluation database, directs the data transfer activities of all field units, and hosts the evaluation web server which provides the caws-evaluaiton.loragen.com dynamic web site. It communicates through a 100 Base-T CAT5 network connection to other local machines (such as a backup computer), and to the field machines via a commercial-class SDSL broadband connection.

5.6.1 Hardware Components

<u>CASE</u>
Type: ANTEC Workstation Tower w/ 400W Supply
Model #: SX1040B

<u>SCSI CARD</u>
Type: Adaptec 29160N SCSI PCI Kit
Model #: 29160N

<u>NETWORK CARD</u>
Type: 3Com 10/100 Secure NIC
Model #: 3CR990-TX-97

<u>CASE FAN (x2)</u>
Type: SUNON 80x80x25mm Case Fan
Model #: KD1208PTB2

<u>MOTHERBOARD</u>
Type: SOYO K7V DRAGON+ Plus VIA KT266A ATX
Model #: SY-K7VDRAGON+

<u>PROCESSOR</u>
Type: AMD ATHLON XP 1800+/1.53 GHz
Model #: AX1800DMT3C

<u>THERMAL COMPOUND</u>
Type: Arctic Silver Thermal Compound
Model #: ARCTIC SILVER III 3G

<u>CPU FAN/HEATSINK</u>
Type: Thermaltake VOLCANO 7 Variable Fan
Model #: A1124

<u>RAM</u>
Type: 512 MB Crucial Micron 64x64 PC2100 DDR RAM
Model #: CT64M72S4D75

<u>HARD DRIVE (x2)</u>
Type: Seagate 36.7GB Cheetah 10K Ultra 160 SCSI
Model #: ST336706LW

<u>CDRW</u>
Type: Plextor PleXWriter 12/10/32 Internal IDE
Model #: PX-W1210TA

<u>FLOPPY DRIVE</u>
Type: Sony 1.44MB Floppy Drive
Model #: MPF920-Z

<u>VIDEO CARD</u>
Type: ATI Xpert 98 Rage XL 2X AGP 8MB

5.6.2 Software Components

5.6.2.1 System Software

Hard drive partitioning. OS, MySQL, PHP, Apache.

Version 1.3.29 of the Apache web server is installed to provide the public access to the collected data.

Version 4.3.4 of the PHP scripting language is installed to provide a means to deliver dynamic content through the Apache web server.

Version 3.23.49-log of the MySQL database engine is installed to store the collected data from the field acquisition computers.

Two SCSI Seagate 36.7 GB hard drives are installed on the central server. These drives are setup in SCSI mirror mode. The same exact data is stored on each hard drive in case of a hard drive failure. In Linux, MD stands for mirrored disk. The following information can be obtained from the "**df**" command.

Table 5.6.1. Central Partition Setup.

			Mounted on
/dev/md3	ext3	380760	/
/dev/md4	ext3	46537	/boot
/dev/md2	ext3	505508	/home
/dev/md1	ext3	2063440	/usr
/dev/md0	ext3	28898044	/var

5.6.2.2 Gather Process

Loragen Script (loragen)

Location

/etc/init.d/loragen

Usage

```
loragen (start | stop)
```

Description

This script is automatically invoked when the server boots. Its main purpose is to start and stop the Loragen gather script file.

Gather Script (gather)

Location

/usr/local/loragen/gather

Usage

gather

Description

This script access the MySQL database on the central server and calls the gather web page for each field data acquisition computer. Recent modifications enable server to gather data from the field data acquisition computers ramdrive and hard drive (used to back up data when the internet connection goes down).

Gather PHP Script (gather.php)

Location

/var/www/caws-evaluation.loragen.com/gather.php

Usage

gather.php?fdb=[db_name]&id=[site_id]

Description

This PHP script transfers all MySQL data from the field computers to CAWS-evaluation server once called by gather script, based on site_id and db_name, and is what allows CAWS-evaluation server to never loose or duplicate data from the field computers. The script does the following steps to retrieve data from the specific field computer called by the gather script; for detailed steps look at gather.php script.

- Defines max_gathers, number of rows to retrieve, for each specific type of data
    - These limits were put in place to make sure no data was loss during the gather process because tables were being locked. The changing of these limits is not advised due to possible loss of data.
- Get IP address of field machine from `site` table in central database
- Get last gathered IDs from table gather_status_field2 or gather_status_fieldbkp depending on db_name specified in calling the script
- Establish a connection with field computer and select correct database
- Gather data from field computers using the following functions
    - gather_fog
    - gather_speed
    - gather_day_or_night
    - gather_image

- gather_sign
- gather_status
- gather_watchdog_data

- The gather functions do the following things
  - Lock field data table
    - Although this prevents new data to be written to that table, data is not lost. MySQL stores update and insert queries until table is unlocked.
  - Select all data from that field computer table after last collection point and limited by max_gather
  - Insert data into central database
  - Delete all data on field computer successfully collected by central database
  - Optimize field computer's table
    - Prevents `id` fields from reaching their limits by resetting auto-increment to 1 if all data has been removed from that table
  - Unlock field data table

- Update field `last_contact` in table `site` with current timestamp

5.6.2.3 Web Server

The Apache web server has been configured to store web pages for the http://caws-evaluation.loragen.com website in **/var/www/caws-evaluation.loragen.com/**. This directory contains the scripts and web pages that have been used to access data throughout the evaluation. Refer to section 5.8 for details on the CAWS Evaluation web site hosted by the CAWS-evaluation server.

5.6.2.4 Backup Scripts

Backup scripts are automatically run on the CAWS-evaluation server and should be put onto removable media. See section 5.10.3 for detailed instructions on removing backed up data to permanent media.

/usr/local/loragen/backup-db [script]

This script is automatically runs every Friday at 11:45PM, through the loragen crontab, and will backup the database. Script is currently disabled due to DB replication done by a Loragen backup server and to conserve disk space on CAWS-evaluation server. Script stores its DB backup at [/var/backup] with name of central.db.[date].sql and should be moved to CD/DVD on a weekly basis to avoid disk space issues on CAWS-evaluation

/usr/local/loragen/backup-images [script]

This script is automatically run every day at 11:30PM, via the Linux periodic scheduler called crontab (specifically the loragen user crontab), and will backup database images after there are more then 165000 images. The script removes the oldest 25000 images using the MySQL command 'mysqldump' and deletes them from the image_file table, leaving only the references to each in the image_data table.

After the image_file table reaches a size specified by the img_threshold variable in this script, a total of number of image files equal to the img_count variable in the script will be removed from the CAWS database. Image files are removed starting with oldest first, progressing toward more recently acquired files.

These two variables can be adjusted by editing the script and changing the variables 'img_count' (number of images to be removed), and 'img_threshold' (maximum number of image files that can be stored on CAWS database before automatic removal).

Dumps are stored in location [/var/backup] under name central.image_file.[start_id].[end_id].sql and should be moved onto CD/DVD every month to avoid disk space issues on CAWS-evaluation. The variables in this script are normally set to assure that the database size is managed below 4GB.

/usr/local/loragen/backup-logs [script]

This script is automatically run every Saturday at 11:45PM, through root crontab, and will backup apache and loragen log files. Log backups will be stored at [/var/backup] as either apache.logs.YYYYMMDD.tar.gz or loragen.logs.YYYYMMDD.tar.gz for apache and loragen logs respectively. All old logs will be deleted after being backup. Log backups should be taken off of CAWS-evaluation and put on CD/DVD every month.

5.6.3 Database Replication

A backup Linux computer running a MySQL database can be setup to perform database replication of the CAWS-evaluation database as an alternative to using the database backup script (SS 5.6.2.4). Consult http://www.mysql.com to set up a slave database on a separate Linux machine. Once the backup computer is setup with MySQL, the table below enumerates the steps required to successfully replicate the entire CAWS-evaluation server database on the backup machine.

Follow Table 5.6.2, line by line, and be conscious of which system is being modified. It is suggested to do this from a remote terminal with a high-resolution screen with two SSH sessions active on each of the two systems. For both systems log in as root and have one session in a command shell and the other in MySQL (activated from a command shell). Make sure that the session that is logged into MySQL is logged out of MySQL before stopping MySQL. All MySQL statements end in ';' otherwise statements are typed in shell.

Table 5.6.2. Step-by-step process for setting up a backup computer to replicate the CAWS database.

CAWS [Master]	BACKUP [Slave]
	• Stop slave replication **slave stop;**
	• Stop MySQL from running **/sbin/service mysql.server stop**
	• Delete old MySQL central DB director **rm –rf /usr/local/mysql/data/central**
• Stop Gather	

<table>
<tr><td>Update site set gather_from='N' where ID<6;</td><td></td></tr>
<tr><td>• Purge log files
flush logs;</td><td></td></tr>
<tr><td>• Wait for gathers to exit
ps –auwx (all 'sleep' should be 300)</td><td></td></tr>
<tr><td>• Stop all web page gathers
service loragen stop</td><td></td></tr>
<tr><td>• Flush tables with read lock
flush tables with read lock;</td><td></td></tr>
<tr><td>• Read the binary log file name and position number and copy down
show master status;<pre>
+------------------------+-----------+
| File | Position |
+------------------------+-----------+
| caws-evaluation-bin.038 | 1358 |
+------------------------+-----------+
</pre></td><td></td></tr>
<tr><td>• Tar the central DB
tar –cPvf snapshot-mysql.tar /var/lib/mysql/central</td><td></td></tr>
<tr><td>• Generate checksum for snapshot
md5sum snapshot-mysql.tar > md5sum.snapshot.txt</td><td></td></tr>
<tr><td></td><td>• Copy over snapshot file and checksum
scp –p loragen@192.168.0.5:/var/backup/snapshot-mysql.tar /var/backup/

scp –p loragen@192.168.0.5:/var/backup/md5sum.snapshot.txt /var/backup/</td></tr>
<tr><td></td><td>• Compare checksum
md5sum snapshot-mysql.tar;cat md5sum.snapshot.txt</td></tr>
<tr><td></td><td>• Unpack tables
tar –xvvf snapshot-mysql.tar</td></tr>
<tr><td></td><td>• Move tables
mv /var/backup/central /usr/local/mysql/data/</td></tr>
<tr><td></td><td>• Start MySQL server w/o SLAVE functionality
/usr/local/mysql/bin/safe_mysqld –skip-slave-start</td></tr>
<tr><td></td><td>• Configure SLAVE
change master to master_log_file='<recorded log file name>', master_log_pos=<recorded log offset>;</td></tr>
<tr><td>• Unlock the tables
unlock tables;</td><td></td></tr>
<tr><td>• Start gather
update site set gather_from='Y' where ID<6;</td><td></td></tr>
<tr><td>• Start all web page gathers
service loragen start</td><td></td></tr>
<tr><td></td><td>• Start SLAVE replication
slave start;</td></tr>
<tr><td></td><td>• Check SLAVE status, make sure pos is changing and no errors
show slave status;</td></tr>
<tr><td colspan="2" align="center">*****If everything is working and all gather scripts are working*****</td></tr>
<tr><td>• Delete tar and txt file that were generated</td><td>• Delete tar and txt file that were copied over</td></tr>
</table>

The following process can also be used to restore an existing replicated database. Some possible reasons for having to restore a database replication are when the backup computer is rebuilt, the backup computer is disconnect from CAWS-evaluation server for a period of time during which log files on CAWS-evaluation have been deleted/flushed, or in the event of a malfunction of the backup computer requiring a system rebuild.

Included in the deliverable software and on CAWS-evaluation server [/var/db/backup-slave-db] is a script that can be added to root's crontab, on the backup server, that will automatically do a mysqldump of the entire backup DB onto a removable disk drive. Use of this script will eliminate the need to use the backup script on CAWS-evaluation server as described in SS 5.6.2.4. To use this script, copy it over to the backup server and log into the backup server as root. Edit the crontab file with **crontab –e** . For example, the following line added to the bottom of the crontab script will run the script every Saturday at 0205, causing the backup of the entire CAWS database onto the backup system (note: backup and slave are synonymous):

05 2 * * 6 root script_location/backup-slave-db

Make sure to also make the following changes to the backup-slave-db file on the backup system:

If the mount drive location is not /dev/sda1, change it to the correct location throughout entire script.

If the folder /backup does not exist, create this folder so that the backup (SCSI on our system) drive has a mount location in the Linux file system.

If the user name and password are not already the same (on the backup system as the CAWS server) in the connection string to MySQL, change the connection string with the proper username and password for slave DB The connection string will look like:

$ mysql –u <username> –p<password>

This script was written and intended to work with a removable SCSI drive, but the script can be modified to allow the use of a permanently mounted drive by commenting out all lines that 'mount' and 'umount' the drive. Be sure that the backup database drive has adequate space available, since each backup file is 4 to 4.5 Gbytes.

Every time (usually once every week on Saturday night) the automatic backup occurs, it writes a new backup image file to the backup system's hard disk, but it does NOT remove old backup files. If there is inadequate space on the disk for adding the new backup file (e.g., 4.5 Gbytes*), the backup attempt will fail*, no warning will be issued. It is therefore imperative that the backup image files on the backup system's hard disk be periodically copied onto removable media such as a DVD, and removed from the backup hard disk. The capacity of the backup system drive will determine how frequently this must be MANUALLY done. We currently use a SCSI backup disk with a formatted capacity of 18 Gbytes, which allows us to do backups to DVD as infrequently as once every three weeks. However, we maintain a rigid schedule of moving the backup file or files (if you miss a week) to DVD once every week.

5.7 Database Structure

5.7.1 Field System Database

The CAWS distributed database is implemented with MySQL, a widely used and well-supported public-domain database language www.MySQL.com . Instances of MySQL are run on all the field machines as well as the server. Data redundancy is maintained to assure against the possibility of data loss. Data is transferred from the MySQL database on each field machine to the server, and deleted from the field system only after positive confirmation of the data transfer.

The CAWS evaluation system distributed database is designed for maximum protection from data loss, both internally on each field machine, in the transfer of data to the central server, and after the data is on the server. Focusing on the field machine itself for the moment:

Each field machine contains three redundant data storage mechanisms:

RAM (PC100 SDRAM) with total capacity of 1 Gbyte.

(Sufficiently large RAM capacity to avoid ever having to push data to the hard disk in normal operation.)

IDE flash drive with 256 Meg capacity.

(High reliability, small capacity.)

IDE hard drive capacity is 20 Gbytes.

(High capacity, low reliability. The disk does not normally spin. It is activated only when necessary to transfer data to or from the parallel database. This maximizes the life of this mechanical component, which is usually the point of failure for typical PCs.)

On boot (from the IDE flash drive), the MySQL database program is loaded into RAM, and run from the RAM. The primary database is then maintained in RAM, and a parallel version of the same database is maintained on the hard disk.

Field2 is the main CAWS database loaded from the flash drive into RAM of the field computers. All data is first entered into this database, except when the memory capacity is exceeded (e.g., due to a loss of communications with the central CAWS server). Fieldbkp is an exact replica of field2 in its structure, but it is only populated with data that is older than five minutes, in the event of growth of the field2 database due to a loss of communications with the central CAWS server. Fieldbkp is physically located on the hard disk of each field unit, which is usually dormant to assure maximum lifetime of this high-failure-rate component. The hard disk is activated and data is transferred into fieldbkp only when necessary, as mentioned above.

The database consists of several *tables*. Each table is described below by listing the exact wording of the MySQL command line which created each of the tables. DDL means Data Definition Language, which is the formal name for the MySQL command and its arguments.

5.7.1.1 field2/fieldbkp

Databases field2 and fieldbkp are the MySQL databases which are exactly the same on all five field computers. Some tables are not utilized at sites where data from a particular sensor may not be present. An example of a table not being utilized is the day or night_data table for the Downing Road field site because that site does not have a day/night sensor. All data is logged in the field2 database unless there is a network outage or power failure. In such an event, the fieldbkp database resident on the hard drive of each field computer is used to retain all data until the next opportunity to reconnect to the server. Please refer to section 0 for more details on this feature.

DDL:

```
CREATE TABLE `config` (
  `component_id` tinyint(3) unsigned NOT NULL default '0',
  `state` tinyint(3) unsigned NOT NULL default '0',
  `source` char(50) default NULL,
  `delay` int(10) unsigned NOT NULL default '0',
  PRIMARY KEY  (`component_id`)
) TYPE=MyISAM
```

5.7.1.2 day_or_night_data

DDL:

```
CREATE TABLE `day_or_night_data` (
  `id` tinyint(3) unsigned NOT NULL auto_increment,
  `stamp` timestamp(14) NOT NULL,
  `day_or_night` enum('D','N') NOT NULL default 'D',
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.3 fog_data

DDL:

```
CREATE TABLE `fog_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `stamp` timestamp(14) NOT NULL,
  `coefficient` float(6,2) NOT NULL default '0.00',
```

```
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.4 image_data

DDL:

```
CREATE TABLE `image_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `camera_type` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `image_file` blob NOT NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.5 parallel_speed_data

DDL:

```
CREATE TABLE `parallel_speed_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `lane` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `msec` smallint(5) unsigned NOT NULL default '0',
  `time_of_flight` mediumint(8) unsigned NOT NULL default '0',
  `classification1` mediumint(8) unsigned NOT NULL default '0',
  `classification2` mediumint(8) unsigned NOT NULL default '0',
  `filtered` enum('N','Y') NOT NULL default 'Y',
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.6 sign_data

DDL:

```
CREATE TABLE `sign_data` (
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `stamp` timestamp(14) NOT NULL,
  `action_code` tinyint(4) NOT NULL default '0',
  `packet` blob,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.7 site

DDL:

```
CREATE TABLE `site` (
  `id` tinyint(3) unsigned NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM
```

5.7.1.8 site_status

DDL:

```
CREATE TABLE `site_status` (
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `stamp` timestamp(14) NOT NULL,
  `rd_free` int(10) unsigned default NULL,
  `rd_used` float(4,1) default NULL,
  `hd_status` tinyint(3) unsigned default NULL,
  `hd_free` int(10) unsigned default NULL,
  `hd_used` float(4,1) default NULL,
  `mem_free` int(10) unsigned default NULL,
  `mem_used` int(10) unsigned default NULL,
  `cpu_idle_time` float(5,2) default NULL,
  `cpu_temp` float(5,2) default NULL,
```

```
  `ups_status` tinyint(3) unsigned default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.9 speed_data

DDL:

```
CREATE TABLE `speed_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `lane` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `msec` smallint(5) unsigned NOT NULL default '0',
  `timer_count` mediumint(8) unsigned NOT NULL default '0',
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.1.10 watchdog_data

DDL:

```
CREATE TABLE `watchdog_data` (
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `stamp` timestamp(14) NOT NULL,
  `data_type` enum('T','B') NOT NULL default 'T',
  `data` varchar(64) default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.2 Central Server Database

The central database **‘central’** runs on the CAWS evaluation server, and is the main database of the CAWS evaluation system. Data is accumulated from all the field sites by action of the **gather.php** script which runs periodically on the CAWS-evaluation server (about once every 30 seconds). Please see SS 5.7.2.4 for details on the data gathering process.

Below are the table definitions in this database, and a description of the relationship between each table and the purpose of each table. As before, these are described by showing the exact wording of the MySQL DDL command that created each table.

5.7.2.1 day_or_night_data

Purpose: Stores the transition from day to night or night to day based on field sites day/night sensors.

Relations:

`site_id` → `id` of `site`

DDL:
```
CREATE TABLE `day_or_night_data` (
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '1',
  `stamp` timestamp(14) NOT NULL,
  `day_or_night` enum('D','N') NOT NULL default 'D',
  PRIMARY KEY  (`id`),
  KEY `site_id` (`site_id`,`stamp`)
) TYPE=MyISAM
```

5.7.2.2 error_table

Purpose: Not currently enabled but used to contact someone, via e-mail, with any site errors that occurred. Due to communication problems between CAWS and field sites when using an unstable connection, like a dial up connection, this function was disabled and removed.

5.7.2.3 fog_data

Purpose: Holds visibility excitation coefficient taken by visibility sensors.

Relations:

`site_id` → `id` of `site`

DDL:
```
CREATE TABLE `fog_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `coefficient` float(6,2) NOT NULL default '0.00',
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`)
) TYPE=MyISAM
```

5.7.2.4 gather_status

Purpose: Table used for debugging purposes with same functions as gather_status_field2 and gather_status_fieldbkp.

DDL:
```
CREATE TABLE `gather_status` (
`site_id` tinyint(3) unsigned NOT NULL default ‘0’,
`fog_id` smallint(5) unsigned NOT NULL default ‘0’,
`speed_id` mediumint(8) unsigned NOT NULL default ‘0’,
```

```
`day_or_night_id` tinyint(3) unsigned NOT NULL default '0',
`image_id` smallint(5) unsigned NOT NULL default '0',
`status_id` smallint(5) unsigned NOT NULL default '0',
`sign_id` tinyint(3) unsigned NOT NULL default '0',
`watchdog_data_id` smallint(5) unsigned NOT NULL default '0',
PRIMARY KEY (`site_id`)
) TYPE=MyISAM
```

5.7.2.5 gather_status_field2

Purpose: Holds id status for each type of data that CAWS is gathering from the field sites in case of communications problems or power failure, during gather process, to prevent duplication of data.

DDL:

```
CREATE TABLE `gather_status` (
`site_id` tinyint(3) unsigned NOT NULL default '0',
`fog_id` smallint(5) unsigned NOT NULL default '0',
`speed_id` mediumint(8) unsigned NOT NULL default '0',
`day_or_night_id` tinyint(3) unsigned NOT NULL default '0',
`image_id` smallint(5) unsigned NOT NULL default '0',
`status_id` smallint(5) unsigned NOT NULL default '0',
`sign_id` tinyint(3) unsigned NOT NULL default '0',
`watchdog_data_id` smallint(5) unsigned NOT NULL default '0',
PRIMARY KEY (`site_id`)
) TYPE=MyISAM
```

5.7.2.6 gather_status_fieldbkp

Purpose: Holds id status for each type of data that CAWS is gathering from the field sites in case of communications problems or power failure, during gather process, to prevent duplication of data.

DDL:

```
CREATE TABLE `gather_status` (
`site_id` tinyint(3) unsigned NOT NULL default '0',
`fog_id` smallint(5) unsigned NOT NULL default '0',
`speed_id` mediumint(8) unsigned NOT NULL default '0',
`day_or_night_id` tinyint(3) unsigned NOT NULL default '0',
`image_id` smallint(5) unsigned NOT NULL default '0',
`status_id` smallint(5) unsigned NOT NULL default '0',
`sign_id` tinyint(3) unsigned NOT NULL default '0',
`watchdog_data_id` smallint(5) unsigned NOT NULL default '0',
PRIMARY KEY (`site_id`)
) TYPE=MyISAM
```

5.7.2.7 image_data

Purpose: Relational table between `image_file` and `site` to give the ability to dump images from database while allowing the ability to see that an image was captured. Table also gives camera type where 3 is a traffic image and 4 is a sign image.

Relations:

`site_id` → `id` of `site`
`image_file_id` → `id` of `image_file`

DDL:

```
CREATE TABLE `image_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
```

```
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `camera_type` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `image_file_id` mediumint(8) unsigned default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`,`camera_type`)
) TYPE=MyISAM
```

5.7.2.8 image_file

Purpose: Holds all the images for the relational table `image_data`. Each image file is stored in blob data type therefore cannot be viewed using command line interface of myself, suggest using CAWS website or external DB editor like EMS MySQL Manager to view images.

Relations: NONE

DDL:

```
CREATE TABLE `image_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `camera_type` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `image_file_id` mediumint(8) unsigned default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`,`camera_type`)
) TYPE=MyISAM
```

5.7.2.9 lane_data

Purpose: Static relational table that holds relationships between sites, lanes and calibration.

Relations:

`site_id` → `id` in `site`
`calibration_id` → `id` in `speed_calibration`

DDL:

```
CREATE TABLE `lane_data` (
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `lane` tinyint(3) unsigned NOT NULL default '0',
  `calibration_id` tinyint(3) unsigned NOT NULL default '1',
  PRIMARY KEY  (`site_id`,`lane`)
) TYPE=MyISAM
```

5.7.2.10 sign_data

Purpose: Relational table with timestamp of CMS activation and message id.

Relations:

`site_id` → `id` in `site`
`sign_file_id` → `id` in `sign_file`

DDL:

```
CREATE TABLE `sign_data` (
```

```
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `sign_file_id` smallint(5) unsigned NOT NULL default '0',
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`)
) TYPE=MyISAM
```

5.7.2.11 sign_file

Purpose: Holds unique CMS activation messages. `file_md5` is used to identify if a CMS message is new and is the md5sum of the message. `description` is not filled in until manually edited through CAWS website with admin privileges where a digital replication of the message is shown and can be verified by images taken of CMS at time of activation.

Relations: NONE

DDL:

```
CREATE TABLE `sign_file` (
  `id` smallint(5) unsigned NOT NULL auto_increment,
  `file` blob,
  `file_md5` varchar(32) default NULL,
  `description` varchar(75) default NULL,
  `action_code` tinyint(3) unsigned NOT NULL default '0',
  PRIMARY KEY  (`id`),
  KEY `file_md5` (`file_md5`,`action_code`)
) TYPE=MyISAM
```

5.7.2.12 site

Purpose: Gives names to each field site, their IP address, the last time a connection was made between CAWS and a field computer, and if CAWS should gather information from that field site. `ip` is automatically obtained from field computers when a field computer initializes and should not be modified unless absolutely necessary. Setting `gather_from` from `Y` to `N` will stop CAWS from gathering data from that specific field computer.

Relations: NONE

DDL:

```
CREATE TABLE `site` (
  `id` tinyint(3) unsigned NOT NULL auto_increment,
  `name` varchar(50) NOT NULL default '',
  `ip` varchar(15) default NULL,
  `last_contact` timestamp(14) NOT NULL,
  `gather_from` enum('N','Y') NOT NULL default 'N',
  PRIMARY KEY  (`id`),
  UNIQUE KEY `name` (`name`)
) TYPE=MyISAM
```

5.7.2.13 site_status

Purpose: Stores field computers following performance data: ram disk free/used space, hard drive's status, hard drive free/used space, memory free/used, cpu idle time, cpu temp, and ups status.

Relations:

\`site_id\` → \`id\` of \`site\`

DDL:

```
CREATE TABLE `site_status` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `rd_free` int(10) unsigned default NULL,
  `rd_used` float(4,1) default NULL,
  `hd_status` tinyint(3) unsigned default NULL,
  `hd_free` int(10) unsigned default NULL,
  `hd_used` float(4,1) default NULL,
  `mem_free` int(10) unsigned default NULL,
  `mem_used` int(10) unsigned default NULL,
  `cpu_idle_time` float(5,2) default NULL,
  `cpu_temp` float(5,2) default NULL,
  `ups_status` tinyint(3) unsigned default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`)
) TYPE=MyISAM
```

5.7.2.14 speed_calibration

Purpose: Holds calibration data for loop detectors to improve speed calculations.

Relations: NONE

DDL:

```
CREATE TABLE `speed_calibration` (
  `id` tinyint(3) unsigned NOT NULL auto_increment,
  `name` varchar(50) default NULL,
  `loop_distance` mediumint(8) unsigned NOT NULL default '0',
  `clock_speed` mediumint(8) unsigned NOT NULL default '0',
  `classification_scalar` float(7,5) NOT NULL default '1.00000',
  `speed_offset` float(7,5) NOT NULL default '0.00000',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM
```

5.7.2.15 speed_data

Purpose: Holds speed data information to be calculated based off any calibration modifications.

Relations:

\`site_id\` → \`id\` of \`site\`
\`calibration_id\` → \`id\` of \`speed_calibration\`

DDL:

```
CREATE TABLE `speed_data` (
  `id` int(10) unsigned NOT NULL auto_increment,
```

```
    `site_id` tinyint(3) unsigned NOT NULL default '0',
    `lane` tinyint(3) unsigned NOT NULL default '0',
    `stamp` timestamp(14) NOT NULL,
    `msec` smallint(5) unsigned NOT NULL default '0',
    `timer_count` mediumint(8) unsigned NOT NULL default '0',
    `calibration_id` tinyint(3) unsigned NOT NULL default '1',
    PRIMARY KEY  (`id`),
    KEY `stamp` (`stamp`,`site_id`,`lane`)
) TYPE=MyISAM
```

5.7.2.16 user

Purpose: Holds user information for logging into http://caws-evaluation.loragen.com/ where the username is `email`, the `password` is the md5sum of the password given and `admin_level` should always be `1` except for users that are able to edit CMS messages and be given `2` for `admin_level`.

Relations: NONE

DDL:

```
   CREATE TABLE `user` (
    `id` tinyint(3) unsigned NOT NULL auto_increment,
    `email` varchar(50) NOT NULL default '',
    `password` varchar(16) binary NOT NULL default '',
    `admin_level` tinyint(3) unsigned NOT NULL default '0',
    PRIMARY KEY  (`id`),
    KEY `email` (`email`),
    KEY `password` (`password`)
) TYPE=MyISAM
```

5.7.2.17 user_log

Purpose: Store information on users that log into http://caws-evaluation.loragen.com/, information could be used to generate statistics on who uses the web interface.

Relations:

`user_id` → `id` of `user`

DDL:

```
   CREATE TABLE `user_log` (
    `id` mediumint(8) unsigned NOT NULL auto_increment,
    `user_id` tinyint(3) unsigned NOT NULL default '0',
    `stamp` timestamp(14) NOT NULL,
    `ip` char(15) NOT NULL default '',
    PRIMARY KEY  (`id`),
    KEY `stamp` (`stamp`)
) TYPE=MyISAM
```

5.7.2.18 watchdog_data

Purpose: Stores watchdog timer card data from each field site.

Relations:

`site_id` → `id` of `site`

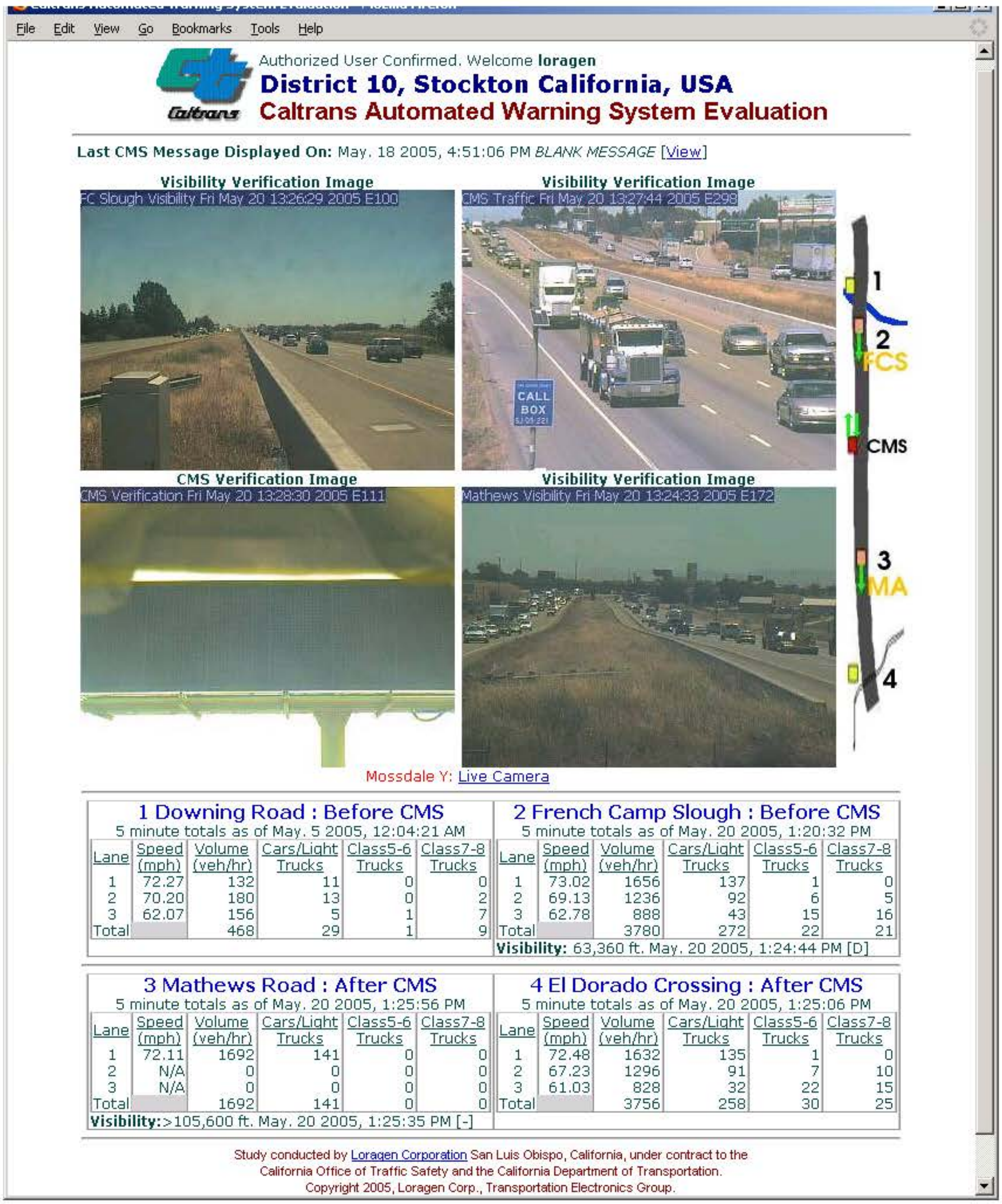
DDL:

```
CREATE TABLE `watchdog_data` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `site_id` tinyint(3) unsigned NOT NULL default '0',
  `stamp` timestamp(14) NOT NULL,
  `data_type` enum('T','B') NOT NULL default 'T',
  `data` varchar(64) default NULL,
  PRIMARY KEY  (`id`),
  KEY `stamp` (`stamp`,`site_id`)
) TYPE=MyISAM
```

5.8 The CAWS Evaluation Web Site

5.8.1 Main Page and Access to Analysis Pages

The only public feature of the CAWS evaluation project is the CAWS Evaluation web site, hosted by the caws server. Its URL is http://caws-evaluation.loragen.com/ .



Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	72.27	132	11	0	0
2	70.20	180	13	0	2
3	62.07	156	5	1	7
Total		468	29	1	9

Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	73.02	1656	137	1	0
2	69.13	1236	92	6	5
3	62.78	888	43	15	16
Total		3780	272	22	21

Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	72.11	1692	141	0	0
2	N/A	0	0	0	0
3	N/A	0	0	0	0
Total		1692	141	0	0

Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	72.48	1632	135	1	0
2	67.23	1296	91	7	10
3	61.03	828	32	22	15
Total		3756	258	30	25

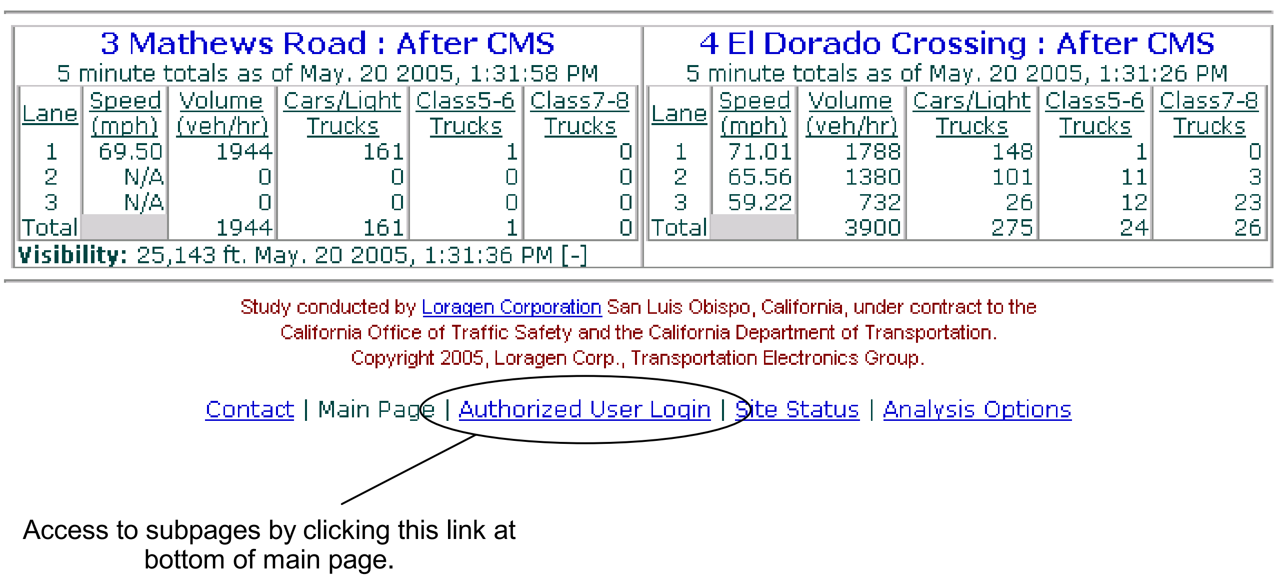
Figure 5.8.1.1. CAWS Evaluation web site main page.

Figure 5.8.1.1 shows the user interface to the main page, with is public-accessible. This page shows, in real time, the current traffic volumes and mean speeds and vehicle class distributions for each lane at each of the evaluation field sites. It also shows the current images from each of the field cameras – one each before and after the CMS, and two at the CMS site – one to verify the CMS message and one to monitor actual traffic and visibility. Visibility sensor readings before and after the CMS are also reported.

Authorized login permits the user access to twenty subpages, each providing a different type of data display and on-line analysis tools. From these pages, it is possible to perform almost any type of analysis of the traffic, visibility, or CMS activity in our evaluation section of the CAWS. At the bottom of the page is a clickable link for authorized user login. The username and password installed for use by Caltrans authorized users is:

Username: caltrans

Password: fogcrash



3 Mathews Road : After CMS

5 minute totals as of May. 20 2005, 1:31:58 PM

Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	69.50	1944	161	1	0
2	N/A	0	0	0	0
3	N/A	0	0	0	0
Total		1944	161	1	0

Visibility: 25,143 ft. May. 20 2005, 1:31:36 PM [-]

4 El Dorado Crossing : After CMS

5 minute totals as of May. 20 2005, 1:31:26 PM

Lane	Speed (mph)	Volume (veh/hr)	Cars/Light Trucks	Class5-6 Trucks	Class7-8 Trucks
1	71.01	1788	148	1	0
2	65.56	1380	101	11	3
3	59.22	732	26	12	23
Total		3900	275	24	26

Figure 5.8.1.2. Bottom of main web page, showing authorized login link.

The administrative password and other high-security information (that permits a user to modify or erase the database) are included in Addendum A.

5.8.2 Files and Features

Each page is dynamically generated by a PHP script, which reads the most recent data entered into the CAWS database from the field sites. The location of the web site files on the CAWS server's file system is

/var/www/caws-evaluation.loragen.com/

The main (public) page is dynamically generated by the PHP script file (shown with full path name):

/var/www/caws-evaluation.loragen.com/main.php

The authorized login page is generated by:

/var/www/caws-evaluation.loragen.com/login.php

Each of the subpages is generated by one of the following PHP files, shown with the full path name below. Each of the subpage PHP files includes detailed comments and instructions on usage and modification.

- /var/www/caws-evaluation.loragen.com/includes/analysis.php
- /var/www/caws-evaluation.loragen.com/includes/common.php
- /var/www/caws-evaluation.loragen.com/includes/db.php
- /var/www/caws-evaluation.loragen.com/analyze_speeds.php
- /var/www/caws-evaluation.loragen.com/dump_images.php
- /var/www/caws-evaluation.loragen.com/dumphist.php
- /var/www/caws-evaluation.loragen.com/login.php
- /var/www/caws-evaluation.loragen.com/php_info_help.php
- /var/www/caws-evaluation.loragen.com/secret-y-camera.php
- /var/www/caws-evaluation.loragen.com/show_image.php
- /var/www/caws-evaluation.loragen.com/show_sign.php
- /var/www/caws-evaluation.loragen.com/show_sign_image.php
- /var/www/caws-evaluation.loragen.com/site_status.php
- /var/www/caws-evaluation.loragen.com/snapshot.php
- /var/www/caws-evaluation.loragen.com/view_all_sites_by_lanes.php
- /var/www/caws-evaluation.loragen.com/view_images.php
- /var/www/caws-evaluation.loragen.com/view_signs.php
- /var/www/caws-evaluation.loragen.com/view_site_status.php
- /var/www/caws-evaluation.loragen.com/view_speeds.php
- /var/www/caws-evaluation.loragen.com/view_visibility.php

5.8.3 Analysis Subpages

To access the subpages after login in, click the Analysis Options link at the bottom of the main page.

This page is generated by /var/www/caws-evaluation.loragen.com/includes/analysis.php. The user will then see the analysis option selection page shown in Figure 5.8.3.1.

Figure 5.8.3.1. Analysis Options Menu Page, CAWS Evaluation Web Server.

Each of the links on this page activates a subpage which provides access to the specified data and associated on-line analysis tools. The bottom of each subpage is identical, providing return navigation links, or direct access to the main page. Each of the analysis option subpages are described below.

5.8.3.1 View all Sites by Lane

Implemented by PHP script view_all_sites_by_lane.php

This page gives a listing of individual lane speeds for each car detected at a given site. The default reference is all vehicles detected just prior to the current time. The site and date and time range can be specified at the top of the page. The duration of the period of observation (span in minutes) can be limited, or the number of displayed records can also be limited. Records can be view by either segregated by lanes, or purely by vehicle time of detection. The standard view example of Figure 5.8.3.2 below shows the table ordered from oldest entry to most recent with only speed displayed.

Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Speed Data

Site	Date					Span	Max.	View
All	05	20	2005	14	13	5	50	Standard

Submit

Displaying 50/442 records from all sites in the 5 minutes from Fri. May. 20 2005, 2:13 PM

Time	Site	Lane	Speed (mph)
2005-05-20 14:13:00.01	El Dorado Crossing	1	72.9
2005-05-20 14:13:00.15	El Dorado Crossing	3	64.4
2005-05-20 14:13:01.44	El Dorado Crossing	1	74.4
2005-05-20 14:13:02.98	El Dorado Crossing	3	78.9
2005-05-20 14:13:03.28	El Dorado Crossing	2	66.0
2005-05-20 14:13:04.05	El Dorado Crossing	2	68.8
2005-05-20 14:13:05.30	El Dorado Crossing	2	74.9
2005-05-20 14:13:06.31	El Dorado Crossing	1	74.4
2005-05-20 14:13:06.48	El Dorado Crossing	3	79.2
2005-05-20 14:13:11.87	El Dorado Crossing	3	54.7
2005-05-20 14:13:12.57	El Dorado Crossing	2	73.0
2005-05-20 14:13:12.94	El Dorado Crossing	1	76.9
2005-05-20 14:13:13.91	El Dorado Crossing	3	64.4
2005-05-20 14:13:14.05	El Dorado Crossing	1	79.4
2005-05-20 14:13:15.68	El Dorado Crossing	2	71.2
2005-05-20 14:13:16.23	Mathews Road	1	76.1
2005-05-20 14:13:16.37	El Dorado Crossing	1	76.9
2005-05-20 14:13:16.74	El Dorado Crossing	3	79.0
2005-05-20 14:13:17.63	Mathews Road	1	73.7
2005-05-20 14:13:18.38	Mathews Road	1	74.8
2005-05-20 14:13:18.44	El Dorado Crossing	1	79.5
2005-05-20 14:13:18.94	El Dorado Crossing	2	91.7
2005-05-20 14:13:19.59	El Dorado Crossing	1	78.5
2005-05-20 14:13:20.16	El Dorado Crossing	2	83.6
2005-05-20 14:13:21.42	El Dorado Crossing	3	64.6
2005-05-20 14:13:21.64	Mathews Road	1	72.1
2005-05-20 14:13:23.26	El Dorado Crossing	2	71.6
2005-05-20 14:13:23.44	Mathews Road	1	70.7
2005-05-20 14:13:24.01	El Dorado Crossing	3	79.1
2005-05-20 14:13:24.87	Mathews Road	1	72.5
2005-05-20 14:13:25.17	El Dorado Crossing	2	72.3
2005-05-20 14:13:27.24	Mathews Road	1	76.5
2005-05-20 14:13:28.04	Mathews Road	1	78.2
2005-05-20 14:13:29.13	El Dorado Crossing	2	69.8
2005-05-20 14:13:29.35	Mathews Road	1	77.7
2005-05-20 14:13:30.78	Mathews Road	1	76.5
2005-05-20 14:13:32.99	El Dorado Crossing	3	64.2
2005-05-20 14:13:35.13	Mathews Road	1	77.3
2005-05-20 14:13:38.85	Mathews Road	1	78.6
2005-05-20 14:13:40.96	Mathews Road	1	76.9
2005-05-20 14:13:41.39	El Dorado Crossing	1	79.0

Figure 5.8.3.2. View all Sites By Lanes - Standard View

The by lanes view shows all lanes at a particular site in a tabular format with detailed information on each vehicle separated into columns for each lane. The site number code is defined on the main page. The separation distance between the car the recorded vehicle and the vehicle in preceding it is also displayed. The vehicle length is displayed as an indicator of the vehicle classification. Also displayed is the day/night, visibility coefficient, visibility distance, and the CMS #1 message at the time of detection. Figure 5.8.3.3 has been modified to show only site four to properly display the image on one page. On the actual web page, all lanes at all sites (a total of 12 lanes) would be displayed if “all” is entered in the Site box.

Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Speed Data

Site: All | Date: 05 - 20 - 2005 | 14 : 35 | Span: 5 | Max.: 50 | View: By Lanes

Submit

Displaying 50/418 records from all sites in the 5 minutes from Fri. May. 20 2005, 2:35 PM

Index	Time of Arrival	S4 Lane 1			S4 Lane 2			S4 Lane 3			Site 1 (FCS)			Site 2 (Mathews)			Message	
		Speed (MPH)	Sep (ft.)	Lngth (ft.)	Speed (MPH)	Sep (ft.)	Lngth (ft.)	Speed (MPH)	Sep (ft.)	Lngth (ft.)	Day/Nite (1/0)	Vis ExCoef.	Vis (ft.)	Day/Nite (1/0)	Vis ExCoef.	Vis (ft.)	Code	Text
1	2005-05-20 14:35:01.27	77.3	N/A	20.63							1	0.20	79200	1	0.36	44000	26	BLANK MESSAGE
2	2005-05-20 14:35:02.11	74.4	66.1	25.54							1	0.20	79200	1	0.36	44000	26	BLANK MESSAGE
3	2005-05-20 14:35:02.32							64.7	N/A	70.36	1	0.20	79200	1	0.36	44000	26	BLANK MESSAGE
4	2005-05-20 14:35:04.43				66.3	N/A	22.98				1	0.20	79200	1	0.36	44000	26	BLANK MESSAGE
5	2005-05-20 14:35:04.78	71.8	260.5	20.62							1	0.20	79200	1	0.36	44000	26	BLANK MESSAGE
6	2005-05-20 14:35:05.83	72.5	90.2	21.43							1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
7	2005-05-20 14:35:06.63										1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
8	2005-05-20 14:35:06.71							64.7	327.6	88.93	1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
9	2005-05-20 14:35:08.18										1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
10	2005-05-20 14:35:08.54							64.7	127.3	46.31	1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
11	2005-05-20 14:35:08.55				62.5	314.4	63.22				1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
12	2005-05-20 14:35:08.85	73.3	298.9	25.78							1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
13	2005-05-20 14:35:09.61										1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
14	2005-05-20 14:35:09.62	73.3	61.4	21.36							1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
15	2005-05-20 14:35:10.26				67.2	149.7	18.8				1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
16	2005-05-20 14:35:11.01				60.9	-10.8	77.82				1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
17	2005-05-20 14:35:11.63	77.7	207.9	21.17							1	0.20	79200	1	0.50	31680	26	BLANK MESSAGE
18	2005-05-20 14:35:13.43				63.1	199.6	24.39				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
19	2005-05-20 14:35:13.52										1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
20	2005-05-20 14:35:14.03				64.2	36.6	19.91				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
21	2005-05-20 14:35:14.54	71.8	285.4	21.07							1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
22	2005-05-20 14:35:14.92				63.6	61.3	21.73				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
23	2005-05-20 14:35:15.59	74.4	90.8	23.76							1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
24	2005-05-20 14:35:16.26										1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
25	2005-05-20 14:35:16.87																	
26	2005-05-20 14:35:17.58				61.4	156	19.64				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
27	2005-05-20 14:35:17.89				62.0	43.7	20.83				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
28	2005-05-20 14:35:19.97	71.0	218.9	20.59							1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
29	2005-05-20 14:35:20.70										1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
30	2005-05-20 14:35:22.32							54.9	919.6	59.49	1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
31	2005-05-20 14:35:22.36										1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
32	2005-05-20 14:35:22.54				62.5	360.5	77.67				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
33	2005-05-20 14:35:22.70							54.5	78.4	68.65	1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
34	2005-05-20 14:35:24.40	81.7	554.9	21.46							1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
35	2005-05-20 14:35:24.42	93.7	210.4	23.23							1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
36	2005-05-20 14:35:26.69							55.0	114	37.69	1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
37	2005-05-20 14:35:27.09							54.7	163.3	18.78	1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
38	2005-05-20 14:35:28.08										1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE
39	2005-05-20 14:35:28.16				66.3	531.7	24.52				1	0.22	72000	1	0.50	31680	26	BLANK MESSAGE

Figure 5.8.3.3. View all sites by lane - By Lanes View

5.8.3.2 View Speed Data

Implemented by PHP script: view_speeds.php

This page gives a listing of speeds for each car at a given site. The reference time is the current server time, which can be changed at the top of the page. Data can be filtered by changing the site, date, time, span (number of minutes ahead of given time), max number of data rows to show and view (Standard or By Lanes). The standard view shows a table from oldest entry to most recent with only speed displayed and looks identical to Figure 5.8.3.2. The by lanes view gives the speed and headway of each individual car recorded. Headway is measured in centiseconds (hundredths of a second) and is the measure of the time separation between the detection of the back of the lead car and the front of the trail car. Figure 5.8.3.4 shows sample data for the El Dorado site. Only one site may be selected for the by lanes view.

Caltrans

Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Speed Data

Site: El Dorado | Date: 05 - 20 - 2005 | 15 : 24 | Span: 5 | Max.: 50 | View: By Lanes

Submit

Displaying 50/355 records from in the 5 minutes from Fri. May. 20 2005, 3:24 PM

Time of Arrival	Lane 1 Speed (MPH)	Lane 1 Hdwy (cs)	Lane 2 Speed (MPH)	Lane 2 Hdwy (cs)	Lane 3 Speed (MPH)	Lane 3 Hdwy (cs)
2005-05-20 15:24:00.99			65.7	N/A		
2005-05-20 15:24:02.59					64.9	N/A
2005-05-20 15:24:03.68			68.5	270		
2005-05-20 15:24:05.74			71.9	217		
2005-05-20 15:24:06.87	76.0	N/A				
2005-05-20 15:24:08.13					64.8	527
2005-05-20 15:24:09.20			65.1	330		
2005-05-20 15:24:10.39					64.8	215
2005-05-20 15:24:12.27	68.6	543				
2005-05-20 15:24:12.38			67.5	315		
2005-05-20 15:24:13.26	69.0	100				
2005-05-20 15:24:13.75					64.9	320
2005-05-20 15:24:14.01			67.2	161		
2005-05-20 15:24:14.08	69.6	84				
2005-05-20 15:24:14.37					65.0	59
2005-05-20 15:24:14.89	68.0	81				
2005-05-20 15:24:15.26			66.6	122		
2005-05-20 15:24:16.37	70.0	152				
2005-05-20 15:24:16.64			66.6	135		
2005-05-20 15:24:17.07					64.8	257
2005-05-20 15:24:19.49			72.6	303		
2005-05-20 15:24:21.35			68.5	187		
2005-05-20 15:24:22.03					79.5	578
2005-05-20 15:24:22.54	71.4	646				
2005-05-20 15:24:23.40	68.6	87				
2005-05-20 15:24:23.43			76.5	233		
2005-05-20 15:24:24.04			72.3	65		
2005-05-20 15:24:24.61	72.5	129				
2005-05-20 15:24:25.46	71.4	89				
2005-05-20 15:24:25.59			66.6	151		
2005-05-20 15:24:26.01					55.0	321
2005-05-20 15:24:27.35	72.5	201				
2005-05-20 15:24:28.30	74.4	104				
2005-05-20 15:24:29.25					54.6	259
2005-05-20 15:24:34.22	77.7	675				
2005-05-20 15:24:34.60			67.8	896		
2005-05-20 15:24:36.79			66.6	214		
2005-05-20 15:24:37.48			68.2	69		
2005-05-20 15:24:37.84					54.7	689
2005-05-20 15:24:38.14	71.0	408				

Figure 5.8.3.4. View Speeds - By Lanes

5.8.3.3 View CMS Data

Implemented by PHP script: view_signs.php

This page displays all CMS messages sent to CMS #1, with the most recent CMS message first. The only active filters to view data are the date and maximum number of messages to display. By default the server time is used as reference. There are three links for each CMS message; view, image and speeds.

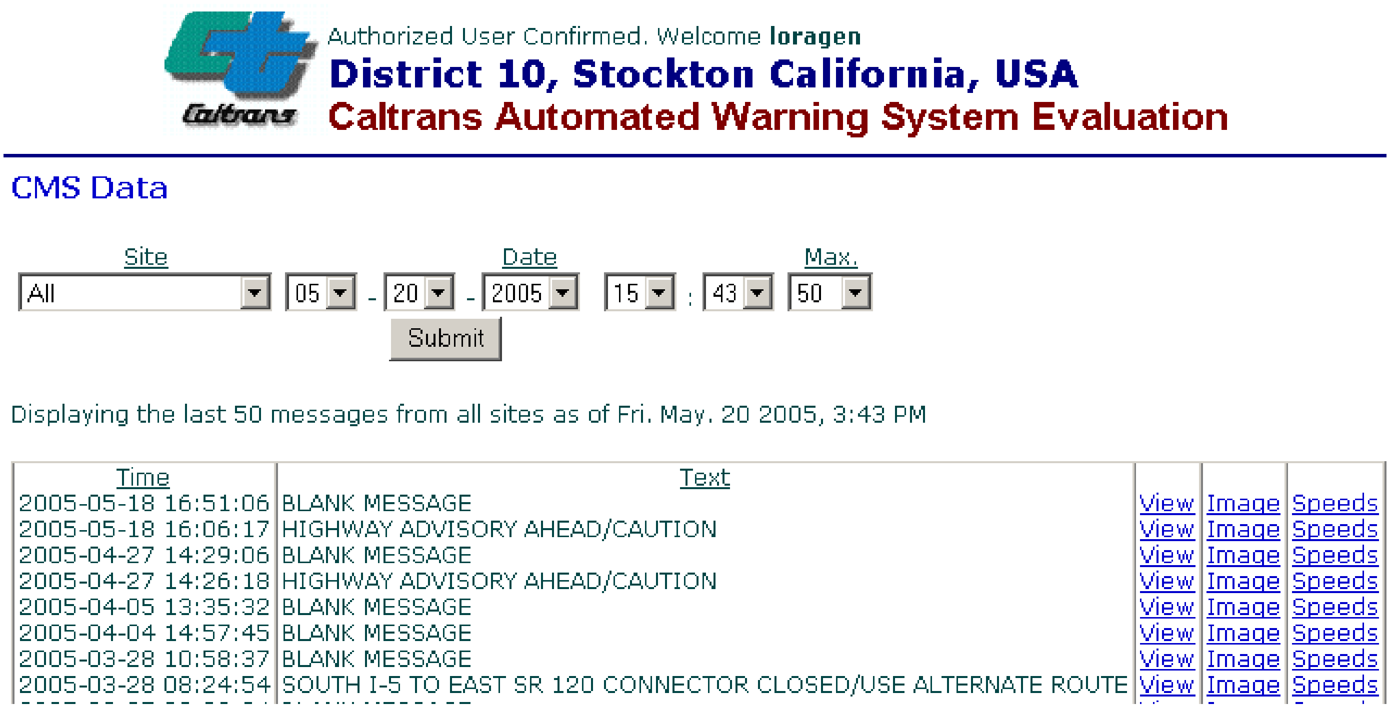


Figure 5.8.3.5. View CMS Data - Main Page

The View link will display the computer-generated image of the CMS message at the time most recent time the message changed. The View link can also be used to change the CMS message Text if not recognized by the CAWS-evaluation server, represented by **<unknown>** in the text column of Figure 5.8.3.5. Note: in order to change CMS message text the user must be logged in with admin privileges.

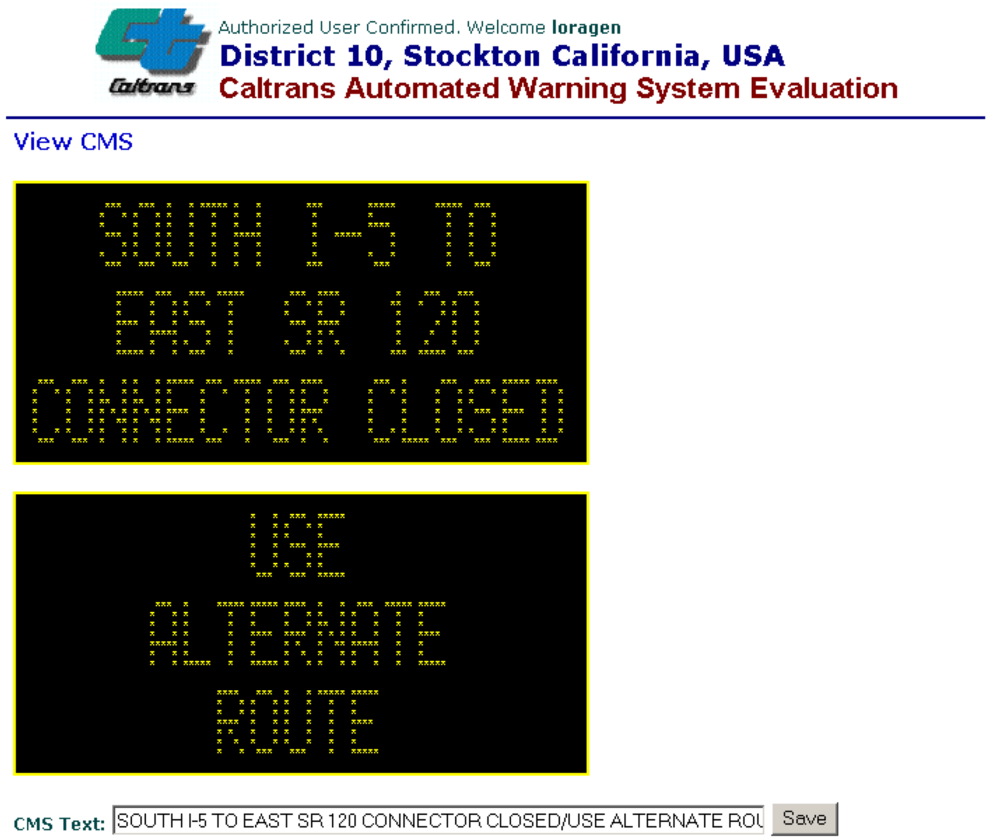


Figure 5.8.3.6. View CMS Data - Computer Generated Message

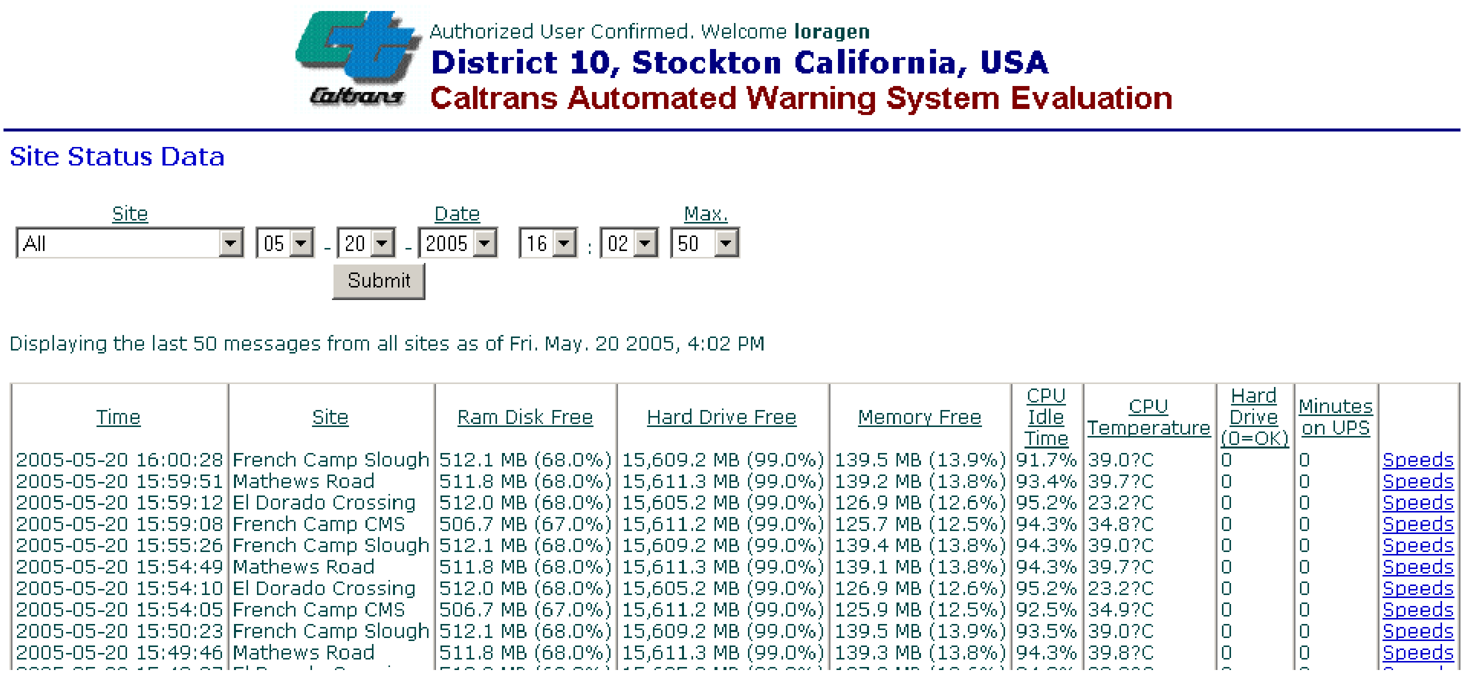
The Image link will open a pop-up window and display the image taken at the time of the CMS message activation. If no pop-up window appears make sure all pop-up blockers are turned off or set to have the CAWS-evaluation website as a trusted site. If an error of “image has been backed up” occurs, these image files may still be access by following the procedure of SS 5.10.4.2 on restoring previously backed up image data.

Figure 5.8.3.7. View CMS Data - CMS Image View

5.8.3.4 View Site Status Data

View Site Status Data (view_site_status.php)

This page will display all system status records reported by the field sites at any selected time or range of times. The table displayed will give the time, site, RAM disk free, hard drive free, memory free, CPU idle time, CPU temperature, hard drive status, and minutes on UPS if running on backup power at the reporting time. The default display time is the current server time and a maximum of 50 records displayed. These values can all be changed. The Speeds link will take you to the standard View Speed Data page with the date/time of the CMS activation as the start time.



Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Site Status Data

Site: All | Date: 05 - 20 - 2005 | 16 : 02 | Max.: 50

Submit

Displaying the last 50 messages from all sites as of Fri. May. 20 2005, 4:02 PM

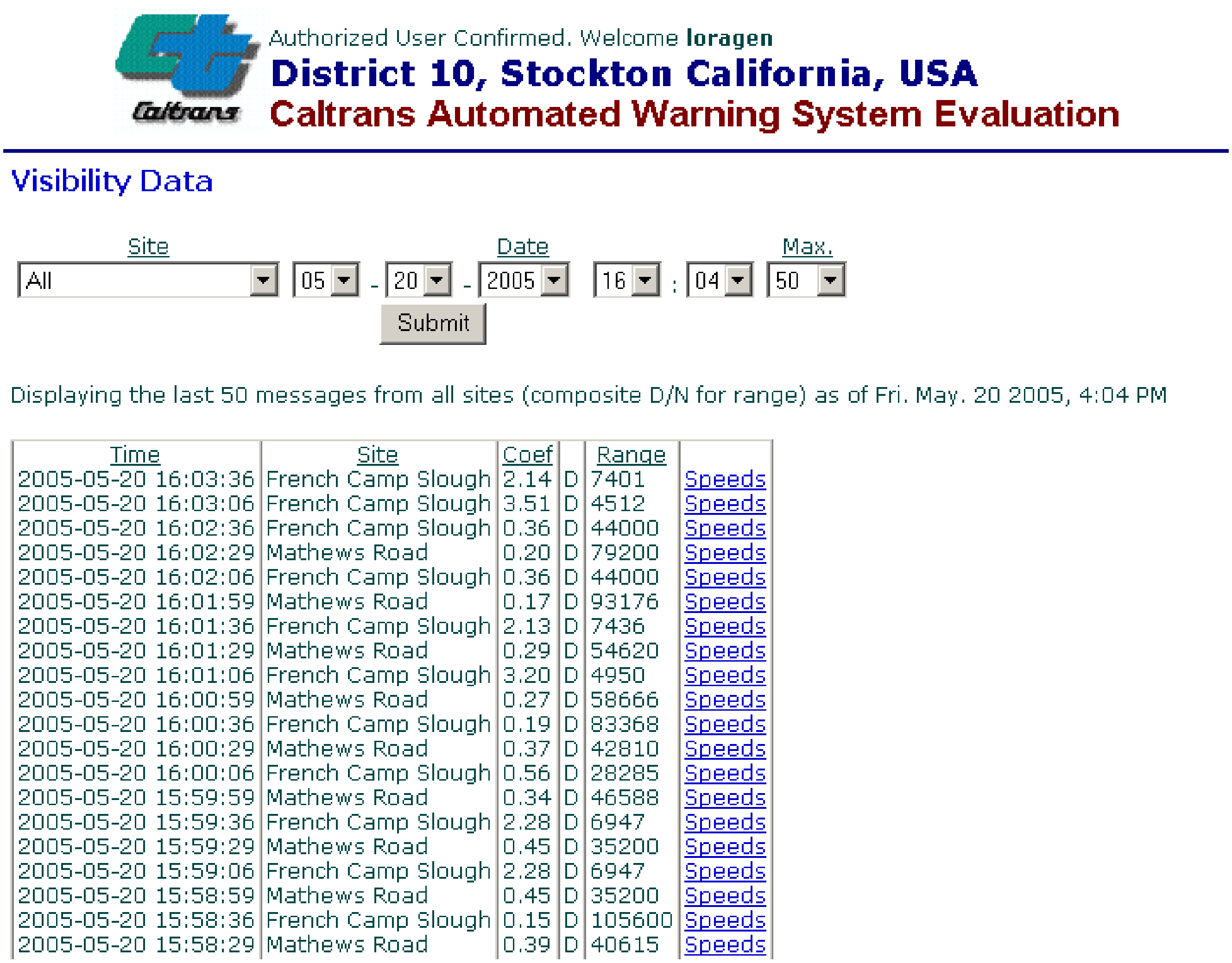
Time	Site	Ram Disk Free	Hard Drive Free	Memory Free	CPU Idle Time	CPU Temperature	Hard Drive (0=OK)	Minutes on UPS	
2005-05-20 16:00:28	French Camp Slough	512.1 MB (68.0%)	15,609.2 MB (99.0%)	139.5 MB (13.9%)	91.7%	39.0?C	0	0	Speeds
2005-05-20 15:59:51	Mathews Road	511.8 MB (68.0%)	15,611.3 MB (99.0%)	139.2 MB (13.8%)	93.4%	39.7?C	0	0	Speeds
2005-05-20 15:59:12	El Dorado Crossing	512.0 MB (68.0%)	15,605.2 MB (99.0%)	126.9 MB (12.6%)	95.2%	23.2?C	0	0	Speeds
2005-05-20 15:59:08	French Camp CMS	506.7 MB (67.0%)	15,611.2 MB (99.0%)	125.7 MB (12.5%)	94.3%	34.8?C	0	0	Speeds
2005-05-20 15:55:26	French Camp Slough	512.1 MB (68.0%)	15,609.2 MB (99.0%)	139.4 MB (13.8%)	94.3%	39.0?C	0	0	Speeds
2005-05-20 15:54:49	Mathews Road	511.8 MB (68.0%)	15,611.3 MB (99.0%)	139.1 MB (13.8%)	94.3%	39.7?C	0	0	Speeds
2005-05-20 15:54:10	El Dorado Crossing	512.0 MB (68.0%)	15,605.2 MB (99.0%)	126.9 MB (12.6%)	95.2%	23.2?C	0	0	Speeds
2005-05-20 15:54:05	French Camp CMS	506.7 MB (67.0%)	15,611.2 MB (99.0%)	125.9 MB (12.5%)	92.5%	34.9?C	0	0	Speeds
2005-05-20 15:50:23	French Camp Slough	512.1 MB (68.0%)	15,609.2 MB (99.0%)	139.5 MB (13.9%)	93.5%	39.0?C	0	0	Speeds
2005-05-20 15:49:46	Mathews Road	511.8 MB (68.0%)	15,611.3 MB (99.0%)	139.3 MB (13.8%)	94.3%	39.8?C	0	0	Speeds

Figure 5.8.3.8. View Site Status Data

5.8.3.5 View Visibility Data

Implemented by PHP script: view_visibility.php

This page displays data taken from the visibility sensor and day/night sensor at each of the two sites equipped with these instruments. The default time is set to current server time and can be changed at the top of the page. Also the maximum number of records can be change. The table reports the time, site, visibility extinction coefficient, day or night value, and FAA visibility range. The visibility coefficient and day/night status are used to calculate the visibility range in feet according to standard formulas.


Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Visibility Data

Site: All | Date: 05 - 20 - 2005 | 16 : 04 | Max. 50

Submit

Displaying the last 50 messages from all sites (composite D/N for range) as of Fri. May. 20 2005, 4:04 PM

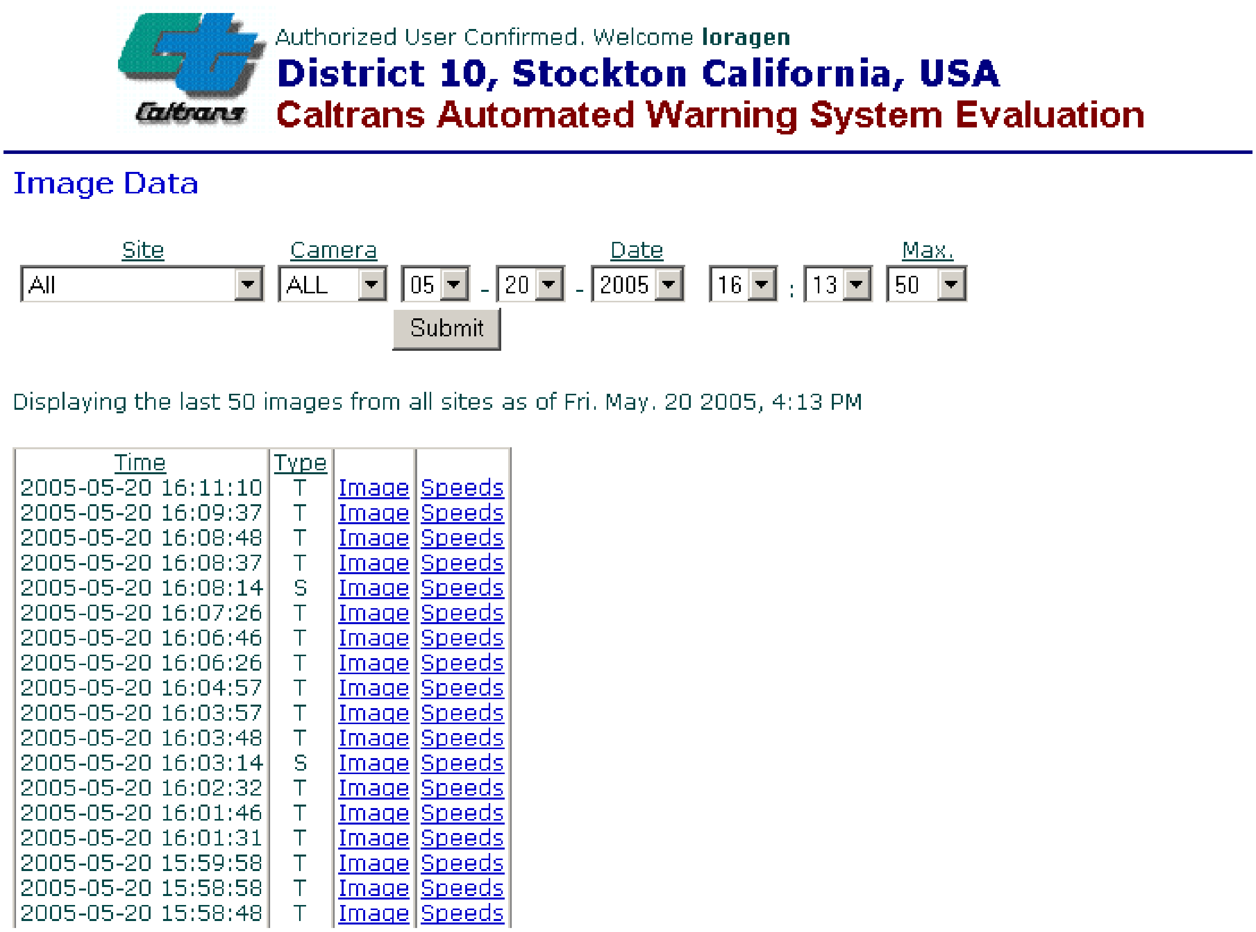
Time	Site	Coef		Range	
2005-05-20 16:03:36	French Camp Slough	2.14	D	7401	Speeds
2005-05-20 16:03:06	French Camp Slough	3.51	D	4512	Speeds
2005-05-20 16:02:36	French Camp Slough	0.36	D	44000	Speeds
2005-05-20 16:02:29	Mathews Road	0.20	D	79200	Speeds
2005-05-20 16:02:06	French Camp Slough	0.36	D	44000	Speeds
2005-05-20 16:01:59	Mathews Road	0.17	D	93176	Speeds
2005-05-20 16:01:36	French Camp Slough	2.13	D	7436	Speeds
2005-05-20 16:01:29	Mathews Road	0.29	D	54620	Speeds
2005-05-20 16:01:06	French Camp Slough	3.20	D	4950	Speeds
2005-05-20 16:00:59	Mathews Road	0.27	D	58666	Speeds
2005-05-20 16:00:36	French Camp Slough	0.19	D	83368	Speeds
2005-05-20 16:00:29	Mathews Road	0.37	D	42810	Speeds
2005-05-20 16:00:06	French Camp Slough	0.56	D	28285	Speeds
2005-05-20 15:59:59	Mathews Road	0.34	D	46588	Speeds
2005-05-20 15:59:36	French Camp Slough	2.28	D	6947	Speeds
2005-05-20 15:59:29	Mathews Road	0.45	D	35200	Speeds
2005-05-20 15:59:06	French Camp Slough	2.28	D	6947	Speeds
2005-05-20 15:58:59	Mathews Road	0.45	D	35200	Speeds
2005-05-20 15:58:36	French Camp Slough	0.15	D	105600	Speeds
2005-05-20 15:58:29	Mathews Road	0.39	D	40615	Speeds

Figure 5.8.3.9. View Visibility Data Subpage.

5.8.3.6 View Images

View Images (view_images.php)

This page displays all camera images acquired by the four cameras in the CAWS-evaluation system. The locations of the four video cameras are identified on the main page. The default display time is the current server time. Data can be filtered based off of time, camera type, and maximum number of records to display. Camera type 'T' indicates a traffic view, while Type 'S' indicates a "Sign Verification" view, for the camera directed at CMS #1. The Images link will open up a pop-up window which will display the image taken at that timestamp. If am error message "Image Backed Up" appears, the image has been removed from the CAWS-evaluation database and backed up to CD. Please refer to Section 5.10.4.2 on details of restoring images to CAWS-evaluation server. The Speeds link will open the View Speed Data page with the corresponding speed records starting at that timestamp.



Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Image Data

Site: All | Camera: ALL | Date: 05 - 20 - 2005 16 : 13 | Max.: 50

Submit

Displaying the last 50 images from all sites as of Fri. May. 20 2005, 4:13 PM

Time	Type		
2005-05-20 16:11:10	T	Image	Speeds
2005-05-20 16:09:37	T	Image	Speeds
2005-05-20 16:08:48	T	Image	Speeds
2005-05-20 16:08:37	T	Image	Speeds
2005-05-20 16:08:14	S	Image	Speeds
2005-05-20 16:07:26	T	Image	Speeds
2005-05-20 16:06:46	T	Image	Speeds
2005-05-20 16:06:26	T	Image	Speeds
2005-05-20 16:04:57	T	Image	Speeds
2005-05-20 16:03:57	T	Image	Speeds
2005-05-20 16:03:48	T	Image	Speeds
2005-05-20 16:03:14	S	Image	Speeds
2005-05-20 16:02:32	T	Image	Speeds
2005-05-20 16:01:46	T	Image	Speeds
2005-05-20 16:01:31	T	Image	Speeds
2005-05-20 15:59:58	T	Image	Speeds
2005-05-20 15:58:58	T	Image	Speeds
2005-05-20 15:58:48	T	Image	Speeds

Figure 5.8.3.10. View Images Subpage.

Selection of the image link will generate an image in a new browser window such as the example of Figure 5.8.3.11 for a traffic (Type T) image, or Figure 5.8.3.12 for a CMS verification (Type S) image.

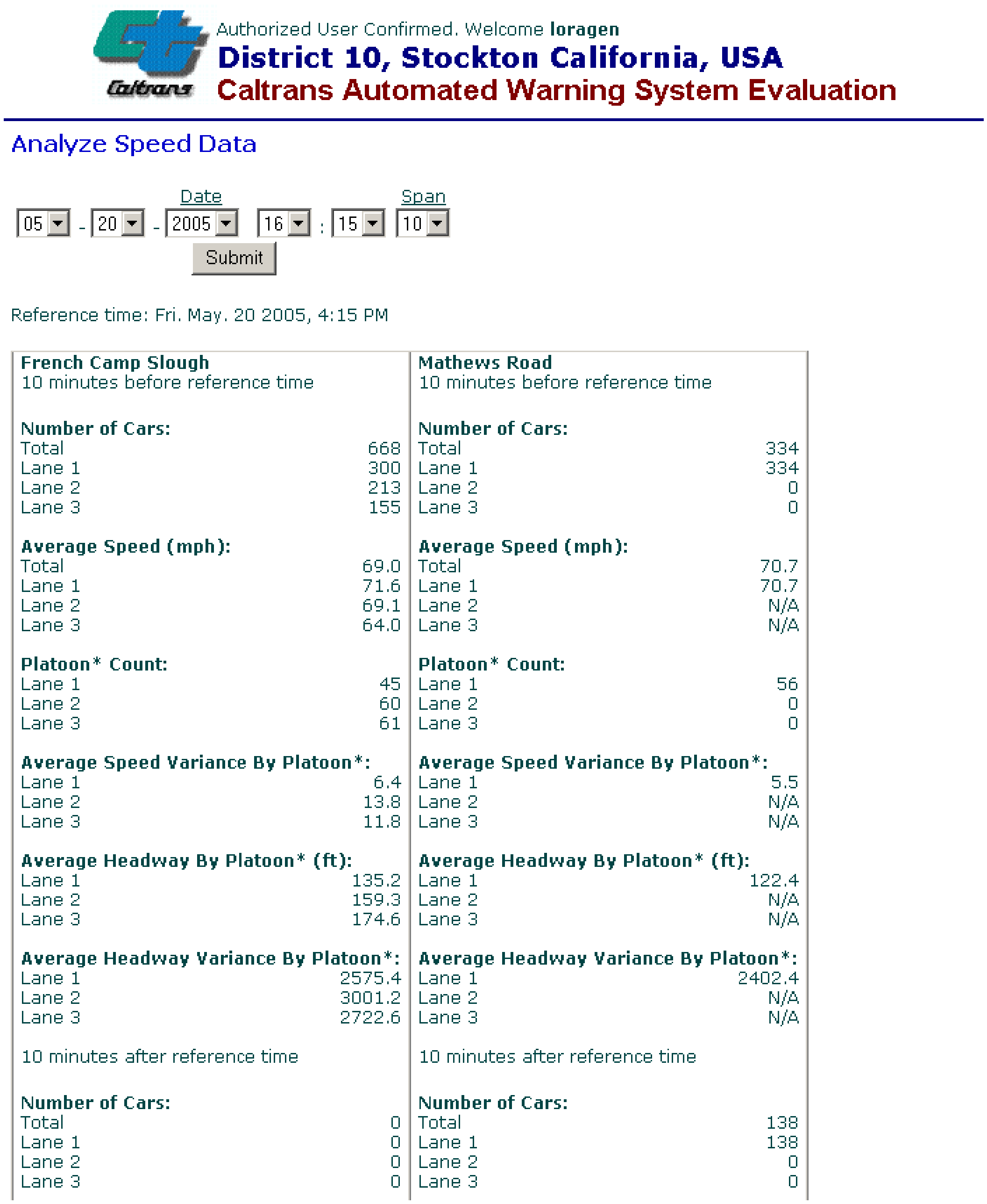
Figure 5.8.3.11. Example of Traffic (Type T) Camera Image from CMS Traffic Monitoring Camera.

Figure 5.8.3.12. (Type S) camera image of CMS message during fog.

5.8.3.7 Analyze Speed Data

Implemented by PHP script: analyze_speeds.php

This page supports the analysis of driver behavior by displaying overall statistics at the sites immediately before and after CMS #1. The default display time is the current server time, which can be adjusted at the top of the page. The 'span' option determines the maximum period for data records after the specified display time. Each data category is broken down for each of the three lanes at the site. The categories are 'Number of Cars', 'Average Speed (mph)', 'Platoon Count', 'Average Speed Variance By



Authorized User Confirmed. Welcome **loragen**

District 10, Stockton California, USA

Caltrans Automated Warning System Evaluation

Analyze Speed Data

Date: 05 - 20 - 2005 16 : 15 Span: 10

Submit

Reference time: Fri. May. 20 2005, 4:15 PM

French Camp Slough		Mathews Road	
10 minutes before reference time		10 minutes before reference time	
Number of Cars:		**Number of Cars:**	
Total	668	Total	334
Lane 1	300	Lane 1	334
Lane 2	213	Lane 2	0
Lane 3	155	Lane 3	0
Average Speed (mph):		**Average Speed (mph):**	
Total	69.0	Total	70.7
Lane 1	71.6	Lane 1	70.7
Lane 2	69.1	Lane 2	N/A
Lane 3	64.0	Lane 3	N/A
Platoon* Count:		**Platoon* Count:**	
Lane 1	45	Lane 1	56
Lane 2	60	Lane 2	0
Lane 3	61	Lane 3	0
Average Speed Variance By Platoon*:		**Average Speed Variance By Platoon*:**	
Lane 1	6.4	Lane 1	5.5
Lane 2	13.8	Lane 2	N/A
Lane 3	11.8	Lane 3	N/A
Average Headway By Platoon* (ft):		**Average Headway By Platoon* (ft):**	
Lane 1	135.2	Lane 1	122.4
Lane 2	159.3	Lane 2	N/A
Lane 3	174.6	Lane 3	N/A
Average Headway Variance By Platoon*:		**Average Headway Variance By Platoon*:**	
Lane 1	2575.4	Lane 1	2402.4
Lane 2	3001.2	Lane 2	N/A
Lane 3	2722.6	Lane 3	N/A
10 minutes after reference time		10 minutes after reference time	
Number of Cars:		**Number of Cars:**	
Total	0	Total	138
Lane 1	0	Lane 1	138
Lane 2	0	Lane 2	0
Lane 3	0	Lane 3	0

Figure 5.8.3.13. Analyze Speed Data

Platoon', 'Average Headway By Platoon', and 'Average Headway Variance by Platoon'. A platoon is considered to be a group of proximate cars in a single lane separated by no more then 300 feet. An example is shown in Figure 5.8.3.13.

5.9 Software Tools to Aid in the Analysis of Traffic and Driver Behavior

In the course of our driver behavior analysis work, we developed a number tools to aid in the rapid extraction of selected data from the CAWS database, and to automatically generate metrics from these data. We interact with the database in the same way that the active web pages of the caws-evaluation.loragen.com web site interact with the database. The interactive scripting language PHP is used, which generates well-formatted dynamic outputs which are viewed with an web browser. Data from the browser display are easily copied in to Excel or other spreadsheet programs for further analysis or graphical display.

Listed below are several PHP scripts, each contained in a file of the same name, which perform specific types of data extraction, analysis and formatting. The source code for each script is included on the CD's delivered with the final evaluation project report. These are also resident on the CAWS server itself, for ready use.

To run any of these PHP scripts, first connect to the caws evaluation server over either a local area network or via an external Internet connection. If connecting externally, it is necessary to use a secure (SSH) connection to the server. Each script is run from a web browser (e.g., Netscape or Internet Explorer) by simply typing the name of the script into web address box/bar.

For example: To examine the events surrounding a CMS activation on May 15, 2005 from 8pm to midnight, use the **globaltimes.php** and **dumpspeeds2.php** PHP scripts, and copy the data into the [date]-[event]-event.xls Excel template. Here's the step-by-step procedure:

Using either an SFTP program like WS_FTP or HTML editor with SFTP capabilities like Dreamweaver, and edit the script **globaltimes.php** which is located at /var/www/caws-evaluation.loragen.com/includes/globaltimes.php to specify the start and stop times for the period of interest:

Change 'GLOBAL_STAMP_START' to '20050515200000'

Change 'GLOBAL_TIME_INTERVAL_SECS' to '1140' which is 4 hours and 15 minutes.

After modifying the **globaltimes.php** script, save the script. If saved locally, be sure to upload back to the server in the original location of the script file, above.

In a web browser such as Netscape or Internet Explorer, type in the following address into the address box/bar:

http://caws-evaluation.loragen.com/dumpspeeds2.php

This runs the script, which will proceed to extract the requested data from the CAWS database, and display it in a format ready for import into Excel for graphing or further analysis. Depending on the time

interval of data to be displayed and the network connection speed to the CAWS-evaluation server, data may take up to 2 minutes to be assembled, downloaded and displayed by the web browser.

Once data is displayed in the web browser, it may be copied and pasted into a pre-formatted Excel spread sheet following the steps outlined in the corresponding SS 5.9.2.1 below.

A nearly identical process is used to run any of the other scripts, by substituting the script name and template name in the procedure above.

5.9.1 Description of PHP Analysis Scripts

All Driver Behavior Analysis [DBA] PHP scripts access [includes/globaltimes.php] where they get the start time to gather their data and duration for how long they gather data. All scripts are in location [/var/www/caws-evaluation.loragen.com/].

5.9.1.1 <u>includes/globaltimes.php</u>

Script sets global variables 'GLOBAL_STAMP_START' and 'GLOBAL_TIME_INTERVAL_SECS'. Each DBA PHP script uses these global variables to determine when to start gathering data and duration of gather.

'GLOBAL_STAMP_START' is in the format of 'YYYYMMDDHHmmSSss'

YYYY – four digit year

MM – two digit month

DD – two digit day

HH – two digit hour

mm – two digit minute

SS – two digit second

ss – two digit millisecond

'GLOBAL_TIME_INTERVAL_SECS' is in seconds, and should be in two hour multiples up to 10 hours with an additional 15 minutes added to work properly with event excel sheet. [date]-[event]-event.xls is developed to handle a maximum of 10 hours 15 minutes of data and minimum of 2 hours 15 minutes of data.

Note: If transitioning between days i.e. going from May 15th to May 16th make sure that a two hour window will end and start at midnight otherwise excel graphs will not work properly for [date]-[event]-event.xls.

5.9.1.2 dumpspeeds2.php

Dumps table with data to be cut and pasted into [data]-[event]-event.xls excel file for information on speed and volume data at sites with data gathered for every 15 seconds. Start copying table from first time row and do not copy headers because they are in excel file.

5.9.1.3 dumpstats.php

Dumps general statistics on the CMS transition effectiveness into table format that can be cut and pasted into [date]-message-transitions.xls

5.9.1.4 dumpstats2.php

Dumps general statistics on CMS effectiveness on a constant message interval of one hour. Table can be cut and pasted into [date]-constant-message-intervals.xls

5.9.2 Description of Excel Template Files

Once data is generated by the action of one of the above PHP scripts and displayed via a web browser, it can be easily copied and pasted into Excel or another spreadsheet of statistical analysis program for further analysis, or more typically, to take advantage of the rich graphing and data rendering capabilities of these programs. The copy and paste operation uses the regular ODBC (Open Database Connectivity) features standard with any MS Windows operating system.

When copying into Excel, it is convenient to copy into an already-formatted spreadsheet, with cells pre-configured to expect the PHP-generated tabular data. Below is a description of a number of pre-formatted spreadsheet templates, and the process by which data should be copied into them.

All template and example files are located at /var/DBA/ on the CAWS server.

5.9.2.1 [date]-[event]-event.xls

This .xls template displays graphical plots of a number of traffic metrics measured before and after the CMS: mean speed, standard deviation of speed, traffic volume, accident risk factor, visibility, % difference of mean speed, and % difference of accident risk factor. The current CMS message is also displayed graphically. The template is populated from data generated by the **dumpspeeds2.php** script described above. Data should be copied from the browser display starting with the first row containing data. Column headers do not need to be copied because they are already in the excel template.

Steps to using this excel file:

1. Run dumpspeeds2.php on server.
2. Copy data from dumpspeeds2.php into cell ‘A3’
3. Find and replace all ‘N/A’ with nothing

4. Find and replace all CMS MSG with appropriate numerical code:

CMS MSG	# Code
BLANK MESSAGE	0
AMBER ALERT	10
DENSE FOG/ADVISE 45 MPH	20
DENSE FOG/ADVISE 30 MPH	30
HIGHWAY ADVISORY AHEAD/CAUTION	40
SLOW TRAFFIC AHEAD/CAUTION	50
STOPPED TRAFFIC AHEAD/CAUTION	60
GUSTY WIND WARNING	70
OTHER	80

5. Change tab names for all graphs using month and date, i.e. may16
6. Modify x-axis scales for each graph [**Note**: must enter in mixed military format, i.e. 5:30pm is 17:30. If going from 10pm to 12am then enter 22:00, 24:00 respectively and for 12am to 2am enter 0:00 and 2:00 respectively]

Delete any tabs that are not being used for specific study.

5.9.2.2 [date]-message-transitions.xls

Steps to using excel file:

1. Copy entire data table form dumpstats.php
2. Delete current table in excel file by right clicking upper left header, which selects all cells, and delete.
3. Paste table from dumpstats.php into cell ‘A1’.
4. Again clicking upper left header, then double-click in between two columns to auto resize and the same for rows.
5. Select all date/times and set vertical alignment to center.

Select each time row using Ctrl to select multiple rows and then select whole table and give a thick surrounding boarder.

5.9.2.3 [date]-constant-message-interval.xls

Steps to using excel file:

1. Follow steps 1-4 for [date]-message-transitions.xls then save.

5.10 System Backup and Provisions for Data Protection

The CAWS-evaluation server provides several automated mechanisms to aid in the backup and removal of data to keep the size of the central database below an acceptable maximum. The maximum database size is limited by two considerations:

1. The available disk capacity (approx 36.7 Gbytes of which approximately 30 Gbytes are allocated for data).

2. The capacity of a typical writable DVD, 4.7 Gbytes.

Data is generated by three main sources on the server:

1. Data collected from the field machines, which is entered directly into the database by the gather process run on the CAWS Evaluation server. This includes traffic data, visibility data, and camera images.

2. By the Apache Web Server that automatically generates log files when recording web server activity.

3. Log files are automatically generated by the gather process, which the CAWS Evaluation server uses to collect data from the field sites. These log files record all data exchange activity, or lack thereof.

5.10.1 Automated Data Backup Mechanisms

The following mechanisms are provided for automated file system and database management. Three automatic scripts previously described in Section 4.2.4 implement these mechanisms.

1. (Optional) Periodic backup of the caws evaluation database to a separate (backup) computer over a local area network onto a removable SCSI drive. The external backup computer initiates this process once a week on Saturday at 2:05 AM.

2. Periodic compression and storage of the web server log files and system data gathering log files into the directory /var/backup/ on the CAWS server. This is done to consolidate these log files in a common convenient directory in preparation for manual backup and removal. This periodic process is initiated by the CAWS Evaluation Server and runs automatically once a week on Saturday at 11:45 PM.

3. Periodic compression and storage of camera image (.jpg) files older than about one month from the caws evaluation database into the directory /var/backup/ on the CAWS server. This is done to keep the size of the active database within manageable limits, since this file is periodically backed up to the external computer (1). This periodic process is initiated by the CAWS Evaluation Server and runs automatically every day at 11:30 PM but only generates a file every 10-12 days.

5.10.2 Required Manual Data Backup Procedures

In addition, certain manual periodic procedures are required for file system maintenance:

1. Permanent storage and removal of the backup database files (created by mechanism 1 above) from the backup computer to removable media such as DVD. (DVD is required since the database file is typically 4 Gbytes, which exceeds the 650 Mbyte capacity of a CD.) The file will be in the format of ‘central.db.[date].sql’ on the SCSI drive of the separate (backup) computer.
2. Permanent storage and removal of web server log files and system data gathering log files from the CAWS Evaluation Server to removable media such as CD. The Apache and gather files will be in the format of ‘apache.logs.[date].tar.gz’ and ‘loragen.logs.[date].tar.gz’ respectively located in the /var/backup/ directory on the CAWS Evaluation server.

3. Permanent storage and removal of camera image files older than about one month from the CAWS Evaluation Server, to removable media such as CD. These files will be in the format of central.image_file.[start_id].[end_id].sql in the /var/backup/ directory.

5.10.3 Removing .jpg Image or Other Data from the Database to Save Disk Space

As discussed above, several automated data backup procedures are implemented on the CAWS server, and some periodic manual backup procedures are required. As data accumulates in the main CAWS database, an increasing amount of disk space is utilized. If automatic backup of the database to an external system is implemented as described in SS 5.7.1, the data in the main database will be duplicated into backup copies on the designated backup system, but no data is removed from the main database. The camera image files (.jpg) that are acquired from the four field cameras take up by far the most space in the database. It is therefore advisable to periodically remove images older than about a month from the database, and back them up to some removable media such as CD or DVD. This frees up space on the server's disk for continued growth of the main database. Other old data in the database cay be removed following the same process. However, traffic and visibility data take up relatively little space in the database, and can be accumulated over many years within the 30 Gbyte capacity of the server's hard dual 30 Gbyte mirrored hard disks.

The procedure below describes how to backup to CD and remove older image data (or other data) from the main database. The CAWS-evaluation server contains a CD writer, which will be used.

Please refer to SS 5.6.2.4 for details on the specific backup scripts called in this process.

On the CAWS-evaluation Server:

Log in as root

Start graphical user interface (GUI)

$ startx

Allow GUI to fully load

Open a 'new terminal'

Move to the directory where backup files are stored

$ cd /var/backup/

Zip the file (.gz) and generate an md5 check sum of file after zipped

$ gzip central.image_file.1635001.1660000.sql

$ md5sum central.image_file.1635001.1660000.sql.gz > md5sum.1635001.1660000.txt

Start CD burning software (X-CD-Roast)

$ xcdroast

A GUI will pop up

‘Create CD’

‘Master Track’

In the ‘File Directory View’ uncheck the “display directories only” check box

Select the files to be burned to CD and click ‘Add’ then ‘OK’

Files

central.image_file.1635001.1660000.sql.gz

md5sum.1635001.1660000.txt

‘Create Session/Image’ tab

‘Calculate Size’

‘Master and Write on-the-fly’

‘OK’

Files will now be burned onto the CD

NOTE: DO NOT CLOSE FRONT COVER OF SERVER

The CD will eject after being burned and could cause damage to CD burner if closed.

Close X-CD-Roast

Now check to make sure data was correctly burned to CD before delete backup files

Mount CD drive

$mount /dev/scd0 /mnt/cdrom

Check files

$md5sum /mnt/cdrom/var/backup/*.gz; cat /mnt/cdrom/var/backup/md5sum*

Check to make sure md5sums are correct

If not then re-burn CD before deleting backup files

Un-mount drive

$umount /dev/scd0

Remove backup files

$rm –rf central.image_file.1635001.1660000.sql.gz md5sum.1635001.1660000.txt

Completely log out of GUI, shell and get to login prompt.

5.10.4 Accessing Archived Camera Images

5.10.4.1 Images Still on CAWS

If images still reside on CAWS-evaluation, go to website and use 'View Images' through 'Analysis Options'. Select date and time and click on 'Image' link and save picture onto HD. If "Error: Image Not Found" is displayed it means this image has been backed up.

5.10.4.2 Recovering Backup Images

Since camera image data takes up much more space on the system hard disk, it is automatically backed up and removed from the database periodically by the backup-images script which runs at 11:30 PM nightly on the CAWS server. The number of image files to be backed up or left in the active database is configurable in the backup script described in detail in SS 5.6.2.4.

If access to image that has already been backed up to CD is required, below are descriptions of two methods for restoring image data from the backup files into either the active CAWS database or another database. There are advantages to each method, discussed below.

5.10.4.3 Method 1

This method does not require user to modify the CAWS-evaluation database, stop gather scripts, or stop services but does require another computer with MySQL installed. The extra computer will be called slave DB and CAWS will be called master DB for this section. In order to view the images the use of some sort of external DB editor like EMS MySQL Manager in order to view the blob images.

Create a DB on slave called 'central' and use following DDL to set up table for images:

```
CREATE TABLE `image_file` (
  `id` mediumint(8) unsigned NOT NULL auto_increment,
  `file` blob NOT NULL,
  PRIMARY KEY  (`id`)
) TYPE=MyISAM
```

Use the following SQL script on master DB to find out what image_file numbers needed:

```
SELECT
  image_data.stamp,
  image_data.image_file_id
FROM
  image_data
WHERE
  (image_data.site_id = 3) AND
  (image_data.stamp BETWEEN '20040930044500' AND '20040930051500')
```

Above will give a list of `image_file_id` with timestamps for dates between 09/30/2004 04:45:00 to 09/30/2004 05:15:00 [times are in military format]. Note the start and end ids from `image_file_id` and place into next script that will be run on slave DB.

Find the backup CD with the image_file ids that are needed. If the image file dump is not zipped the sql file can be read directly off the backup CD but if the file was zipped, it must unzipped and place on the slave HD.

- Windows
    - use winrar to unzip file and extract to temp folder
- Linux
    - mount cdrom drive
    - copy central.image_file.[start_id].[end_id].sql.gz to temp folder
    - unmount cdrom drive
    - gunzip central.image_file.[start_id].[end_id].sql.gz

Open a command window and use following command:

%mysql –u root –p central < [file_location]/central.image_file.[start_id].[end_id].sql

Enter MySQL root password and the process will take 5-15 depending on the speed of the computer. If no error messages occur then process was completed successfully.

If restoring image files out of order then the following SQL command must be entered.

ALTER TABLE image_file ORDER BY `id`

The process will take up to 20 minutes depending on DB size but is done to restore order of image numbers. For example if image_file.id 5-10 were restored before image_file.id 1-4 when doing a search on the image_file table for image_file.id between 4-7 it will only output image_file.id 4.

Once process is finished run the following statement with the correct start and end ids that are needed:

```
SELECT
  image_file.id,
  image_file.`file`
FROM
  image_file
WHERE
  (image_file.id BETWEEN 1055457 AND 1055547)
```

Find image and save to HD.

5.10.4.4 Method 2

This method requires the stopping of all gathering scripts from getting data from field computers. Looking at Table 5.5.3, this process can only be done for less then 127 consecutive days before new data will be lost from field computers. Another limitation is the fact that at 2300 every day the backup_images crontab will run and if backed up images are left on central there will be multiple backed up image dumps and could cause problems with table integrity. Therefore in order to avoid any problems this entire method should be completed in one workday where Method 1 can be completed over multiple workdays.

First all gather scripts must be stopped by running the SQL query:

Update site set gather_from='N' where ID<6;

Check services to make sure gather scripts have stopped by looking for any process that have gather.php and all entries of 'sleep' should be 300.

$ps –auwx

Next stop all the web page gathers:

$/sbin/service loragen stop

Load the backed up images to central DB:

mysql –u root –p central < [file_location]/central.image_file.[start_id].[end_id].sql

Restore structure of image_file table for proper indexing [process may take 20 minutes or longer]:

ALTER TABLE image_file ORDER BY `id`

Use web interface to get all the images needed. Once finished getting all images the backed up images that were restored need to be deleted otherwise at 2300 the backup_images crontab script will create a duplicate mysqldump of images that were restored.

DELETE FROM image_file WHERE (image_file.id BETWEEN 30001 AND 55000)

Last thing is to start the gather scripts again. First run the SQL script:

UPDATE site SET gather_from='Y' where ID<6;

Then start the loragen service:

$/sbin/service loragen start

Check the website after a couple minutes to see if the gather scripts have actually started by seeing if there is recent data.

5.11 System Maintenance, Recovery and Calibration

5.11.1 The Field Unit Boot Sequence

Boot sequence:

System is powered on

System BIOS searches for boot sector on boot media (/dev/hda1 on Flash IDE drive).

System finds boot sector on the Flash IDE drive Partition #1

System loads the boot loader GRUB from the partition (2 stage process)

Boot loader “boot straps” the initial small ramdisk (located on the Flash IDE Drive)

Initial small ramdisk script file **linuxrc** is executed – giving complete control of how OS loads

The Flash and Hard disk partitions are checked and set to autocorrect any file system errors (and reboot if necessary)

The script uncompresses the large ramdisk filesystem (located on the Flash IDE Drive) into ram

The script extracts the modification tar files (from /boot/mods/ directory of Flash IDE Drive) on top of the large ramdisk base filesystem

Initial ramdisk script terminates and the initial (small) ramdisk is unloaded from memory

System resumes the initialization process from the large ramdisk filesystem

5.11.2 Steps to update the Linux kernel

dev - refers to the actual development machine

target - refers to the field machine (where the new kernel will be installed)

To Cross Compile The Kernel, first run:

dev:/usr/src/[linux]/make xconfig

or

dev:/usr/src/[linux]/make config

This will start a program that will ask questions about what needs to be compiled into the kernel for the new build. Once finished answering all the questions, save the new **.config** file somewhere (someone will need to be able to see what's been compiled into the kernel eventually). A good location is **/boot/Config-your-new-kernel-version**

Next, run:

dev:/usr/src/[linux]/make dep

This will make all the kernel dependencies [takes a minute or so].

Now, run:

dev:/usr/src/[linux]/make bzImage

This creates the new kernel image file and system map file.

Next, run:

dev:/usr/src/[linux]/make modules

This creates any modules chosen during the config program. This must be run even if no modules are included.

Make a backup copy of dev:/lib/modules/[linux] (the current version). The next command will overwrite that directory, so be cautious.

dev:/usr/src/[linux]/make modules_install (beware this will overwrite /lib/modules/[linux])

Copy dev:/usr/src/[linux]/System.map --> target:/boot/System.map

Copy dev:/usr/src/[linux]/arch/i386/boot/bzImage --> target:/boot/vmlinuz-[descriptive version]

Copy dev:/lib/modules/[linux-custom...] --> target:/lib/modules/[linux-custom...]

To update the lilo boot loader on the target machine:

Make sure that /boot is mounted on target machine (/dev/hda1).

Make sure that /dev/hda5 is also mounted on the target machine.

Edit the boot loader information:

target:$ vi /etc/lilo.conf

#make image created the default if desired...

#add something like the following

```
image=/boot/vmlinuz-2.4.18-10  #the kernel that will load initially to ram.
    label=i9-2.4.18-10         #descriptive of new kernel and initrd image.
    ramdisk=786432
    initrd=/boot/rdimg9.gz     #don't normally change this.
    root=/dev/ram1
    append="init=/linuxrc"
```

5.11.3 Steps to update the Ramdisk

Have a large partition mounted so the ramdisk can be uncompress. This large partition is referred to here as /backup.

target:$ gunzip -c < /where_ramdisk_image_is/2001.10.11.img.gz > /backup/uncompressed_image_file/new_ramdisk.img

target:$ mount /backup/uncompressed_image_file/new_ramdisk.img /mnt -o loop=/dev/loop0

[this command mounts the ramdisk image to /mnt so that files can be modified and added...etc]

When done modifying the ramdisk, umount it:

target:$ umount /backup/uncompressed_image_file/new_ramdisk.img

Recompress the image file and store it (or overwrite the old one):

target:$ gzip -c9 < /backup/uncompressed_image_file/new_ramdisk.img > /where_ramdisk_image_will_be/new_ramdisk.img.gz

Perform a file system check on the initial ramdisk and the ramdisk image to ensure the integrity.

*e2fsck.ext2 or e2fsck.ext3

Don't perform a bad block test on the file systems unless absolutely needed to.

5.11.4 Creating a swap file

dd if=/dev/zero of=/tmp/my.swap bs=1024 count=65536

[input output blocksize(bytes) count=numberOfBlocks]

This will create a consecutive 64MB swap file

/sbin/mkswap /tmp/my.swap

/sbin/swapon /tmp/my.swap

5.11.5 Basic TAR File Commands for file system maintenance

Tar file creation:

tar -cPf tarfilename.tar file1 file2 file3

(create, preserve permissions, filename, included files)

Tar file addition:

tar --append --file=tarfilename.tar addthis addthat

Tar file deletion:

tar --delete --file=tarfilename.tar toastthis toastthat

Tar single file extraction:

tar --extract --file=tarfilename.tar extractthisfile

5.11.6 How to Install a New Software Module

The CAWS-evaluation server contains the libraries and include files necessary to create new software modules. The “loragen.h” include file contains MySQL prototypes for insertion into the database. The database access routines should always be included from this main file.

The new module must be compiled on the CAWS Evaluation Server to ensure backward compatibility with GCC compatible libraries and routines. It should be possible to compile modules on different Linux computers as long as GCC library version 2.96 and GLIBC (common library) version 2.2.4 is installed.

Loragen.h

After successfully compiling the file, it must be archived into a module. A module is simply a TAR file. The TAR file must be constructed to keep absolute path names. This is necessary so that files are placed in the correct location when the TAR file is "untarred."

tar –cPf modulename.YYYYMMDD.tar /absolute/path/to/fileinclude1 /absolute/path/to/fileinclude2

This command creates the tar file *modulename.YYYYMMDD.tar*.

In order to load the new module onto the ramdisk during system boot, copy the module file onto the boot partition of the flash disk. The first partition of the flash disk contains a directory named "mods." Copy new TAR file into this directory and it will be automatically unpacked onto the ramdisk during system boot.

Note that files will be unpacked from this directory in approximately alphabetical order. Please be aware that the dash character (-) may not be handled correctly in alphabetical order.

5.11.7 How to Configure a Disk Drive on a Field Unit

If a new field unit needs to be constructed, flash drive replaced or hard drive replaced then the use of the Norton utility Ghost can be used to configure the drives. Follow the steps below to create a Ghost image or restore a Ghost image. Pay special attention to which image is being restored.

Restore Image

1) Boot system with Ghost utility from the image CD.
2) Make sure to identify which image is being restored, HD or Flash
3) Options
   a) "Image/Tape"
      i) "Image Boot" for HD
      ii) "Image Disk" for Flash
4) "Local" → "Disk" → "From Image"
5) Open Disk Image
6) Select correct drive
   a) Note drive size and which image is being restored
7) Take CD out and restart system after image is done.

Make Image

1) Boot Ghost utility from bootable floppy made from Ghost.
2) Make sure to identify which image is being restored, HD or Flash
3) "Options"
   a) "Image/Tape"

i) “Image Boot” for HD
ii) “Image Disk” for Flash

4) “Local” → “Disk” → “To Image”

5) Select drive number
   a) HD will have larger size then Flash

6) Select CD/DVD RW drive

7) “Compress Image”
   a) “High” for HD
   b) “No” for Flash

8) “Copy a bootable floppy to the CD/DVD disk?” → “Yes”

9) “Is the floppy disk ready in drive a?” → “Yes”

10) Proceed even if it gives warning of spanning to other CDs, due to compression HD image should fit on one disk.

11) Restart system after image is done or burn other drive, make sure to change “Options”.

5.11.8 How to Replace a Field Unit

Note the cable connections on the field unit that is about to be replaced. Make sure to write down the correct location of the serial port connections. These connections are the easiest to confuse and system will not work properly if connected incorrectly.

Unscrew and remove the old field unit from the rack. Insert the new field unit into the rack and screw it in completely. A monitor and keyboard are required to configure the new field unit. The new field unit has to be configured for the correct site (1-5).

Follow the steps below to configure the field unit for the correct site:

Connect the monitor and the keyboard to the new field unit

Power on the field unit

Select the **root=/dev/hdc2** boot option from the GRUB boot loading menu

Wait for the computer to boot

Login to the computer as the “root” user

Mount the first partition of the flash drive (the boot partition)
$ mount /dev/hda1 /boot

Change the current directory to the boot partition contents
$ cd /boot

List the unique site modules
$ ls modules

Copy over the correct unique site # module to the modules directory
$ cp unique-[1-5].YYYYMMDD.tar /boot/modules/.

Check to make sure that only the one correct unique module is configured

$ ls /boot/modules

Change directories to the root partition

$ cd /

Unmount the boot partition

$ umount /dev/hda1

Force reboot the computer

$ /sbin/shutdown –r now

Wait for the computer to boot

Let the computer automatically select the entry option from the GRUB boot loading menu

When the computer boots it should present the prompt with the correct site number

loragen-field-[1-5] login:

Remove the monitor and keyboard carefully and secure the cabinet

5.11.9 Calibrating the Inductive Loop Detectors for Exact Speed Measurements

5.11.9.1 Calibrating or Recalibrating the Loop Separation Distance

In the event that new inductive loop detectors are installed at a field site, it is necessary to recalibrate the separation distance between each inductive loop in a duplex pair to assure that the reported vehicle speeds are accurate. While the nominal loop separation is typically 20 feet (240 inches), wide variations have occurred in most loop installations. Also, the calibration distance is the inductive distance, which is not necessarily the same as the physical distance between the loops. The best method for calibrating the loop separation is by local direct measurement of the speed of a vehicle crossing both loops. This requires access to some very accurate means for determining the speed of an individual vehicle. A RADAR or LIDAR (light detection and ranging) speed measurement device is preferred.

This procedure requires two people on-site. One person uses the RADAR or LIDAR gun to measure the speed of an isolated vehicle as it crosses over the duplex loops, and another person interacts with the MySQL database, preferably on-console at the field site. Select vehicles separated by at least one full second, but preferably several second, to make it easier to find the vehicle in the database at a later time.

To set the appropriate constants in the database, manually connect to the database on the field machine and display the vehicle speed information in real time as it is detected. Step-by-step, the process is as follows:

Login to the field machine.

Run the MySQL database client program

$ mysql –u root –p field2

Enter the password when prompted

Execute an SQL statement

mysql> SELECT * FROM parallel_speed_data WHERE lane=<lane number> ORDER BY stamp DESC limit 25;

You'll see a text tabular display like the following, for which <lane number> was specified as <1>.

```
+----+------+----------------+------+---------------+----------------+----------------+----------+
| id | lane | stamp          | msec | time_of_flight | classification1 | classification2 | filtered |
+----+------+----------------+------+---------------+----------------+----------------+----------+
| 13 |    1 | 20041123140810 |  542 |        199167 |         230316 |         225581 | N        |
|  9 |    1 | 20041123140808 |   41 |        203049 |         232322 |         223673 | N        |
|  7 |    1 | 20041123140806 |  764 |        204012 |         220863 |         222701 | N        |
|  5 |    1 | 20041123140805 |  803 |        171524 |         195028 |         195982 | N        |
|  3 |    1 | 20041123140802 |  211 |        229732 |         274320 |         276985 | N        |
+----+------+----------------+------+---------------+----------------+----------------+----------+
13 rows in set (0.00 sec)
```

Hit the Up Arrow key on the keyboard to scroll through the command history

mysql> SELECT * FROM parallel_speed_data WHERE lane=<lane number> ORDER BY stamp DESC limit 25;

Execute the same command again to get the most recent data.

For each vehicle that you have measured with the LIDAR gun, record three things:

The site and lane number.

The speed in mph reported by the LIDAR gun for each vehicle, to a precision of 0.1 mph if possible.

The time_of_flight and stamp (time of detection) from the table above. time_of_flight has units of microseconds (10^{-6} seconds) and stamp is the exact date and time of detection in the format of YYYYMMDDHHMMSS (MM/DD/YYYY HH:MM:SS). It will be necessary to refresh this table display immediately after that passage of the vehicle by hitting the Up Arrow and a carriage return to run the same command (as in step 5 above) again.

Repeat this for several vehicles, and also record the site and lane number for each (each lane is different).

The units of the "time_of_flight" and classification data are microseconds (10^{-6} seconds). The conversion of time_of_flight to vehicle speed in miles per hour follows the formulas below:

$$\text{Speed (MPH)} = \frac{\text{Loop Separation (in)}}{\text{Time Of Flight } (\mu\text{s})} \times \frac{1{,}000{,}000\mu s}{1s} \times \frac{60s}{1m} \times \frac{60m}{1hr} \times \frac{1ft}{12in} \times \frac{1mile}{5280ft}$$

$$\text{Speed (MPH)} = \frac{\text{Loop Separation (in)}}{\text{Time Of Flight } (\mu\text{s})} \times 56818.18$$

From these relationships, it is possible to calculate the effective inductive loop separation distance in inches:

$$\text{Loop Separation (in)} = \frac{\text{Recorded Speed } (MPH) \times \text{Time of Flight } (\mu s)}{56818.18}$$

For the vehicles for which you recorded data in a given lane, calculate the inductive loop separation distance in inches using this relationship. Take the average of all of the calculations for each vehicle in the given lane. This is the correct loop separation distance.

The rest of this process requires access to the CAWS Central Server database, which is difficult and slow to access from the field site. This step is best done at the location of the server.

Log into the Central Evaluation server, and access the database by:

Run the MySQL database client program
$ mysql –u root –p central

Enter the password when prompted

The value of the loop separation (in inches) is stored in the CAWS-evaluation central database table speed_calibration. Execute the following SQL statement to correctly identify which speed_calibration id corresponds to a site and lane pair:

mysql>SELECT * FROM lane_data;

Write down the calibration_id for the correct site/lane pair.

Execute the following SQL statement on the central database to identify the current loop_distance separation:

mysql>SELECT * FROM speed_calibration WHERE id=<calibration_id>;

If the loop_distance calculated from your field data for that lane is different than the loop_distance in the table, change the entry in the speed_calibration table by typing the following SQL statement:

mysql>UPDATE speed_calibration SET loop_distance='<loop distance in inches>' WHERE id=<id number>;

After the table has been updated, access the CAWS-evaluation web site and try to verify that the calculation is working properly. Use the 'View Speed Data' analysis option and set the appropriate site, date, and time, and select the 'by lanes' view option. The vehicle time of detection for each of the vehicles you measure at the field site may be used to identify the exact vehicle. Compare the speed now reported by the database with the speed you measured at the site; they should match exactly.

5.11.9.2 Adding a New Lane Entry in the speed calibration table

If any work is done to the physical location of the loops in the road it will be necessary to create entirely new entries for that site in the speed_calibration table and change the relation in the lane_data table on the central database. Do the following steps to complete this process:

Add a new entry for that Site and Lane in speed_calibration table

mysql>INSERT INTO speed_calibration (`name`,`loop_distance`,`clock_speed`,`classification_scalar`,`speed_offset`) VALUES ('<descriptive calibration name>','<loop distance in inches>', '<clock speed>', '<classification scalar>', '<speed_offset>');

After entering new entries into speed_calibration table modify the relational table

UPDATE lane_data SET calibration_id=<new calibration id> WHERE site_id=<site id> AND lane=<lane number>;

Once completed, every new speed entry into the central database gathered from the field computers will have a new calibration_id associated with it and will not corrupt earlier data's calibration_id and loop_distance settings.

5.11.10 How to determine if one of the two mirrored drives on the CAWS server has failed.

As previously discussed, the CAWS server incorporates two redundant SCSI hard drive, which mirror each other at all times to assure maximum protection of data. It's a good idea to check this each time a periodic backup of the images from the data base is done, e.g., once a week.

Type in the following command as root user:

cat /proc/mdstat

If the output looks like this then MD is ok:

```
Personalities : [raid1]

read_ahead 1024 sectors
md4 : active raid1 sdb1[1] sda1[0]
     48064 blocks [2/2] [UU]

md0 : active raid1 sdb2[1] sda2[0]
     29358720 blocks [2/2] [UU]

md1 : active raid1 sdb3[1] sda3[0]
     2096384 blocks [2/2] [UU]

md2 : active raid1 sdb5[1] sda6[0]
     521984 blocks [2/2] [UU]

md3 : active raid1 sdb6[1] sda7[0]
     393472 blocks [2/2] [UU]

unused devices: <none>
```

The critical information above is the [2/2] for each logical drive listing. If it says [2/1], then one of the two mirrored SCSI disks has failed or has partition errors.

For example, If the output from the command # cat /proc/mdstat looks like what you see below, then one of the mirrored drives has failed:

```
Personalities : [raid1]

read_ahead 1024 sectors
md4 : active raid1 sdb1[1]
     48064 blocks [2/1] [UU]

md0 : active raid1 sdb2[1]
     29358720 blocks [2/1] [UU]
```

```
md1 : active raid1 sdb3[1]
      2096384 blocks [2/1] [UU]

md2 : active raid1 sdb5[1]
      521984 blocks [2/1] [UU]

md3 : active raid1 sdb6[1]
      393472 blocks [2/1] [UU]

unused devices: <none>
```

5.11.11 Relocating or renaming the CAWS Evaluation Server

If/when the CAWS evaluation Central Server's external name is changed, modification to an entry in the field computer scripts must be made so that the field computers will be able to report their fixed IP address (assigned by the CDPD service provider) to the Central Server's DB. The process is straightforward:

Server Modifications:

Log into server as root

The external name must be changed in the apache config file, on the server, located at [/usr/local/apache/conf/httpd.conf], use Vim to edit this file.

Find the two locations where 'ServerName' is defined and change the external name to the new external name.

Save the file

Restart the apache service

$/usr/local/apache/bin/apachectl restart

Log out

Field Modifications:

Log into field computer as root

Edit the setMyIP script $vim /usr/local/bin/setMyIP

enter the new external name for the server on line where the url is defined.

url='new_external_address.com'

save the file

Mount the flash drive

$mount /dev/hda1 /boot

Create a tar file with the setMyIP script, place on flash drive, and file will be in format of loragen.20040522.tar ; it is important that the date is the date the file is created because when the system boots it un-tars files from the top of the mods directory alphabetically.

$tar –cPf /boot/mods/loragen.YYYYMMDD.tar /usr/local/bin/setMyIP

Unmount the flash drive

$umount /dev/hda1

Restart the field computer

$/sbin/shutdown –r now

Let the system restart on it's own and make sure that communications for that field computer has been established with server by checking the servers website for new data entries.

5.11.12 Periodic Maintenance and Repair

Only a few periodic hardware and software maintenance procedures are required for the CAWS Evaluation System components. A suggested maintenance interval is included for each item:

The auxiliary deep cycle battery at each field site is a Marine Deep-Cycle Group 24 or Group 27 12 volt lead acid battery. The water level in each cell of the battery should be checked and topped off approximately once a year. The UPS units in each cabinet also contain an internal 12 volt gel cell. This battery requires no maintenance, but has a useful life of about five years.

The All-Weather Systems visibility sensors at the French Camp Slough and the Mathews Road field sites require periodic maintenance according to a schedule set by the manufacturer. The windows of the detectors and emitters must be cleaned (Windex or other window cleaner) at least every three months. The sensor itself should be recalibrated once every six months, or sooner if any form of repair or significant movement has occurred. The calibration is extremely sensitive to the alignment of the emitter with the detector in each of the two pairs.

Each DAS field computer contains a filter behind the front panel, through which all force-air cooling air is passed. It should be cleaned in soapy water at least once a year, or more often in the event of dusty conditions.

Each field cabinet should be inspected approximately once every six months for rodent or water intrusion and/or damage. Failures have occurred due to rodents which chew wires and insulation, and seek the warmth of the computer main board, and water intrusion, which can lead to component failure due to excessive humidity.

Data backup and database maintenance:

Section 5.10 provides information on automatic and required manual periodic data backup and maintenance procedure for the Central Server: We strongly advise implementing the automatic backup to an external system of the database, following the procedure described in Sections 5.10 and 5.7.1.1. If automatic backup is implemented, backup and removal of the database from the external system to removable media is required once a week, or whatever the capacity of the backup computer's hard disk may permit.

In addition, to prevent excessive growth of the main CAWS evaluation database, the older .jpg image files should be removed from the database and stored on CD or DVD at least once a month. The procedure for doing this is given in Section 5.10.3. Each time, check the available space on the CAWS mirrored disked to assure that adequate space remains for growth of the database during the next maintenance interval.

The highest failure rate component of any computer system is the hard disk, which is the only mechanical component. Because of special provisions to maximize hard disk life on the field machines, discussed in Section 5.7.1, no specific replacement interval is advised, although it is advisable to check the physical condition of the hard disk and the motherboards occasionally, especially with regard to humidity-related damage.

Unlike the field machines, the hard disks on the Central Server run continuously and are therefore subject to failure in the manufacturer's recommended time interval. Two 36.7 Gbyte SCSI disk drives are used to store data redundantly by mirroring each other at all times. This assures that if one disk crashes, the other will survive with the entire system image and database intact. If this should occur, immediately replace both disks. Use a disk-image-copying program such as Symantec Ghost to copy the image from the one surviving disk onto the new disks.

We fabricated the server using the highest reliability SCSI hard disks available. The choice of the relatively small 30 Gbyte (redundantly formatted) capacity for each of the mirrored disks was also driven by maximizing reliability. We have never encountered a failure of a disk on the server in approximately 4 years of operation. Hard disks running many years without failure are common, but disk crashes in as little as one year are possible. The life of the hard disk is most directly related to its operating temperature. We therefore strongly advise a constant temperature environment for the CAWS server, and suggest replacement of the disk drives every three years, or whatever the manufacturer's recommended interval may be for a replacement disk drive.

5.12 System Configuration and Network Information

5.12.1 Central Server

External name: caws-evaluation.loragen.com

Operating system: Redhat Linux Release 7.2 (Enigma)

External IP Address: 66.122.64.161

HTTP Server: Apache Web Server 1.3.29

MySQL Version: 3.23.49

Note that software releases are in source form instead of RPM binary format

5.12.2 D10 Weather Server

External Name: sv10tmcweather
Location: Caltrans District 10 Stockton TMC
Internal IP Address: 10.80.11.247

5.12.3 D10 Field Sites

				Phone Line
1	French Camp Slough	loragen-field-1	166.133.71.213	N
2	Mathews Road	loragen-field-2	166.204.2.201	Y
3	French Camp CMS	loragen-field-3	166.130.0.18	Y
4	Downing Road	loragen-field-4	166.133.193.72	N
5	El Dorado Overcrossing	loragen-field-5	166.204.0.102	N
6	Spare Unit	loragen-field-	xxx.xxx.xxx.xxx	

Operating system: Redhat Linux Release 7.1 (Seawolf)
Database software: MySQL Version 3.23.36
SSH is accessible on port 23

5.12.4 Field Computer BIOS Settings

All field computers utilize ABIT model VH6 II motherboards. After considerable testing these mainboards were been found to be the most reliable for 24/7 unattended operation. While no keyboard or monitor is required for remote operation, it is necessary to connect a keyboard and monitor to make system setting changes. Use the "DEL" key to enter the AWARD BIOS setting screen. It is recommended to not change any field computers BIOS settings due to problems that could arise from changes to the serial or parallel port settings. Below are the working default options, in the event that a motherboard must be replaced, or a different motherboard must be substituted.

All source code listings and binaries are included on an Addendum CD to this section of the final project report.

Table 5.12.1. Field Computer Hardware Specs and BIOS Settings.

	750 (100)		SDRAM 10ns
	CPU default		SDRAM 10ns
	Host CLK		SDRAM 10ns
	4 level		Auto
			Auto
	1.44 M, 3.5in		3
	None		Disabled
	Disabled		Enabled
	EGA/VGA		Disabled
	No Errors		Disabled
	Disabled		Disabled
	Enabled		64M
	Enabled		Disabled
	Enabled		Auto
	Disabled		No Support
	Enabled		Disabled
	CDROM		Disabled
	Floppy		Enabled
	HDD-0		Enabled
	Enabled		Enabled
	Disabled		Enabled
	Disabled		Disabled
	On		Disabled
	Enabled		Enabled
	30		Disabled
	250		Disabled
	Setup		1
	Non-OS2		Disabled
	NO		Yes
	Disabled		Disabled

	Disabled		Auto (ESCD)
	Disabled		Disabled
	Disabled		Enabled
	Disabled		Enabled
	Disabled		Auto
	Disabled		Auto
	0		Auto
			Auto

	User Defined		Enabled
	Disabled	-	Auto
	Disabled	-	Auto
	Disabled	-	Auto
	S1 (POS)	-	Auto
	Yes		Enabled
	Suspend → Off	-	Auto
	V/H Sync + Blank	-	Auto
	NA	-	Auto
-	Delay 4 Sec	-	Auto
	On		Enabled
			AGP
	OFF		Disabled
	LPT/COM		Enabled
	ON		Enabled
	OFF		3F8/IRQ4
	Disabled		2F8/IRQ3

	Disabled	-	Disabled
	Disabled		378/IRQ 7
			ECP+EPP
	ON	-	3
	Primary	-	EPP1.9
	Primary		Disabled
	Primary		
	Primary		
	Primary		
	Disabled		
	Secondary		
	Secondary		
	Secondary		
	Primary		
	Primary		
	Primary		
	Disabled		

5.12.5 Central Application Configuration Options

Below are listed complete specifications for all software installed on the central server. All source code listings and binaries are included in appendices to this manual.

PHP Version 4.3.4		
System	Linux caws-evaluation.loragen.com 2.4.20-24.7 #1 Mon Dec 1 13:17:43 EST 2003 i686	
Build Date	Dec 9 2003 16:54:02	
Configure Command	'./configure' '--with-mysql=/usr/local/mysql/' '--with-apxs=/usr/local/apache/bin/apxs' '--with-gd' '--with-jpeg-dir=/usr/lib/' '--with-png-dir=/usr/lib/' '--with-xpm-dir=/usr/X11R6/lib/' '--with-zlib-dir=/usr/include/' '--with-freetype-dir=/usr/ilb/'	
Server API	Apache	
Virtual Directory Support	disabled	
Configuration File (php.ini) Path	/usr/local/lib/php.ini	
PHP API	20020918	
PHP Extension	20020429	
Zend Extension	20021010	
Debug Build	no	
Thread Safety	disabled	
Registered PHP Streams	php, http, ftp, compress.zlib	
Configuration		
PHP Core		
Directive	**Local Value**	**Master Value**
allow_call_time_pass_reference	Off	Off
allow_url_fopen	On	On
always_populate_raw_post_data	Off	Off
arg_separator.input	&	&
arg_separator.output	&	&
asp_tags	Off	Off
auto_append_file	*no value*	*no value*
auto_prepend_file	*no value*	*no value*
browscap	*no value*	*no value*
default_charset	*no value*	*no value*
default_mimetype	text/html	text/html
define_syslog_variables	Off	Off
disable_classes	*no value*	*no value*
disable_functions	*no value*	*no value*
display_errors	Off	Off

display_startup_errors	Off	Off
doc_root	*no value*	*no value*
docref_ext	*no value*	*no value*
docref_root	*no value*	*no value*
enable_dl	On	On
error_append_string	*no value*	*no value*
error_log	*no value*	*no value*
error_prepend_string	*no value*	*no value*
error_reporting	2047	2047
expose_php	On	On
extension_dir	./	./
file_uploads	On	On
gpc_order	GPC	GPC
highlight.bg	#FFFFFF	#FFFFFF
highlight.comment	#FF9900	#FF9900
highlight.default	#0000CC	#0000CC
highlight.html	#000000	#000000
highlight.keyword	#006600	#006600
highlight.string	#CC0000	#CC0000
html_errors	On	On
ignore_repeated_errors	Off	Off
ignore_repeated_source	Off	Off
ignore_user_abort	Off	Off
implicit_flush	Off	Off
include_path	.:/usr/local/lib/php	.:/usr/local/lib/php
log_errors	On	On
log_errors_max_len	1024	1024
magic_quotes_gpc	Off	Off
magic_quotes_runtime	Off	Off
magic_quotes_sybase	Off	Off
max_execution_time	30	30
max_input_time	-1	-1
open_basedir	*no value*	*no value*
output_buffering	4096	4096
output_handler	*no value*	*no value*
post_max_size	8M	8M
precision	14	14
register_argc_argv	Off	Off
register_globals	Off	Off
report_memleaks	On	On
safe_mode	Off	Off
safe_mode_exec_dir	*no value*	*no value*
safe_mode_gid	Off	Off
safe_mode_include_dir	*no value*	*no value*
sendmail_from	me@localhost.com	me@localhost.com

sendmail_path	-t -i	-t -i
serialize_precision	100	100
short_open_tag	On	On
SMTP	localhost	localhost
smtp_port	25	25
sql.safe_mode	Off	Off
track_errors	Off	Off
unserialize_callback_func	*no value*	*no value*
upload_max_filesize	2M	2M
upload_tmp_dir	*no value*	*no value*
user_dir	*no value*	*no value*
variables_order	GPCS	GPCS
xmlrpc_error_number	0	0
xmlrpc_errors	Off	Off
y2k_compliance	Off	Off
apache		
APACHE_INCLUDE	*no value*	
APACHE_TARGET	*no value*	
Apache Version	Apache/1.3.29 (Unix) PHP/4.3.4	
Apache Release	10329100	
Apache API Version	19990320	
Hostname:Port	caws-evaluation.loragen.com:80	
User/Group	apache(48)/48	
Max Requests	Per Child: 0 - Keep Alive: on - Max Per Connection: 100	
Timeouts	Connection: 300 - Keep-Alive: 15	
Server Root	/usr/local/apache	
Loaded Modules	mod_php4, mod_setenvif, mod_so, mod_auth, mod_access, mod_alias, mod_userdir, mod_actions, mod_imap, mod_asis, mod_cgi, mod_dir, mod_autoindex, mod_include, mod_status, mod_negotiation, mod_mime, mod_log_config, mod_env, http_core	
HTTP Headers Information		
HTTP Request Headers		
HTTP Request	GET /php_info_help.php HTTP/1.1	
Accept	image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/msword, application/vnd.ms-excel, */*	
Accept-Encoding	gzip, deflate	
Accept-Language	en-us	
Connection	Keep-Alive	
Host	192.168.0.5	
User-Agent	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0; .NET CLR 1.1.4322)	
HTTP Response Headers		
X-Powered-By	PHP/4.3.4	
Keep-Alive	timeout=15, max=100	

Connection	Keep-Alive	
Transfer-Encoding	chunked	
Content-Type	text/html	
mysql		
MySQL Support	**enabled**	
Active Persistent Links	7	
Active Links	7	
Client API version	3.23.49	
MYSQL_MODULE_TYPE	external	
MYSQL_SOCKET	/tmp/mysql.sock	
MYSQL_INCLUDE	-I/usr/local/mysql//include/mysql	
MYSQL_LIBS	-L/usr/local/mysql//lib/mysql -lmysqlclient	
Directive	**Local Value**	**Master Value**
mysql.allow_persistent	On	On
mysql.connect_timeout	60	60
mysql.default_host	*no value*	*no value*
mysql.default_password	*no value*	*no value*
mysql.default_port	*no value*	*no value*
mysql.default_socket	*no value*	*no value*
mysql.default_user	*no value*	*no value*
mysql.max_links	Unlimited	Unlimited
mysql.max_persistent	Unlimited	Unlimited
mysql.trace_mode	Off	Off

5.12.6 Username and Passwords

Username and Passwords will be constantly changing until the final hand-off to Caltrans District personnel, anticipated to occur in mid-June, 2005. Due to security concerns, this information is contained in a separate document**, Addendum A**, which is provided to a designated Caltrans system manager.

5.13 Evaluation System Transfer Procedures and Issues

At the termination of the contract between the Caltrans and Loragen, ownership and responsibility for the entire CAWS evaluation system will revert to Caltrans. This section covers in detail the procedures required for the transition of operational responsibility with minimal interruption in service, assuming that Caltrans will wish to maintain operation of this system indefinitely as an advanced field data collection and monitoring adjunct to the CAWS.

Currently, the CAWS evaluation server and its associated backup server are located in the Caltrans District 10 TMC. In addition, five field systems are located in the CAWS study area, and the CAWS weather server is located in the District 10 TMC. Loragen uses CDPD and CDMA data 24/7 communications to maintain contact between the field components and the central server. For this purpose, Loragen is under monthly contracts for 5 CDPD unlimited data channels with AirLink Communication of Cupertino, California. It also is under contract with AT&T for one unlimited CDMA data channel. If Caltrans wishes to continue operation of the CAWS evaluation system, they will have to take over responsibility for all these data communications contracts, which will automatically terminate at the end of the Loragen contract with Caltrans. Contact information for each communications service provider is in section 5.4.3.

The CAWS weather server is fully operational in the Caltrans District 10 TMC, and full responsibility assumed by Caltrans. Since it is only accessible only via the Caltrans internal Intranet, and is not accessible by Loragen, full responsibility for this system has been the responsibility of District 10 personnel since 2002, although Loragen continues to service it on-site whenever requested by the District. Separate operating procedures were provided at the time of deliver, which cover all maintenance, backup and accessibility issues for this internal web server, which displays a dynamic map of weather and traffic conditions in the CAWS for internal view of Caltrans staff only.

There are no special requirements for the transition of responsibility for any of the field units; they are designed to be self-sufficient, and as long as they are maintained, they will continue to operate indefinitely, communicating with the CAWS evaluation server.

Here's a checklist of transition issues that must be attended to if uninterrupted operation if the CAWS is to be maintained following the termination of the Loragen contract:

A physical location for the server must be designated. The location must have Internet access with bi-directional bandwidth of at least 384 kbps (commercial SDSL or better).

The five existing CDPD contracts with AirLink Communications must be renewed or replaced with another class of datacom service. With the phase-out of CDPD service in California in favor of CDMA or GPRS, we advise migrating to one of these classes of service. This transition will be facilitated by Airlink, which supports all three types of services. The transition from CDPD to either CDMA or GPRS will require replacement of the stand-alone wireless modems located in each of the five field cabinets. Airlink has announced a buy-back/replacement plan for CDPD modems that Caltrans may wish to take advantage of. The start-up string we use for the CDPD modems will have to be re-written for whatever new type of

modem/service Caltrans selects; the service provider's tech support personnel will assist with this, but it will require the involvement with a person at Caltrans sufficiently knowledgeable with cellular-based wireless data services.

The server must be physically relocated to the designated new Caltrans location, unless Caltrans wishes Loragen to continue to operate the server under a subsequent service agreement. In any case, Loragen will assist with this move and re-activation when requested by Caltrans.

The evaluation URL caws-evaluation.loragen.com should be renamed and re-registered in the dot.ca.gov domain. Several internal IP address will have to be changed also. An entry in the files on each field unit must then be changed to point to the newly named server, and to correctly identity each field unit if their IP addresses are change following the transition to another type of telecom service. This procedure is described in previous sections.

5.14 Technical Support and Contact Information

At the termination of the evaluation contract, Caltrans my elect to continue operation of the field data collection systems. Following the termination of the evaluation contract, all responsibilities for the continued operation of the CAWS evaluation system revert to Caltrans. This document, is intended to provide Caltrans with all information necessary to permit them to operate, duplicate or modify the present system in support of the evaluation of other active safety-enhancement systems or traffic studies. In addition, Loragen will remain available to assist with any questions, recommendations, or services needed to facilitate the transition of the system to successful operation by Caltrans for a reasonable period of time. It is our desire to see that this very advanced data collection remain in operation indefinitely, providing valuable information not only for traffic safety assessment and management, but also in support of the transportation research community.

Contact information:

Primary contacts: Art MacCarley, Tabber Watts

Phone: 805 781 8461

Email: caltrans_support@loragen.com

Address: Loragen Corporation, 3576 Empleo Street, Unit 1, San Luis Obispo, California, 93401