Lauritzen Inc.

Bringing new ideas to renewable energy

Controllers to Valhalla Server Events and Alarms

1. INTRODUCTION

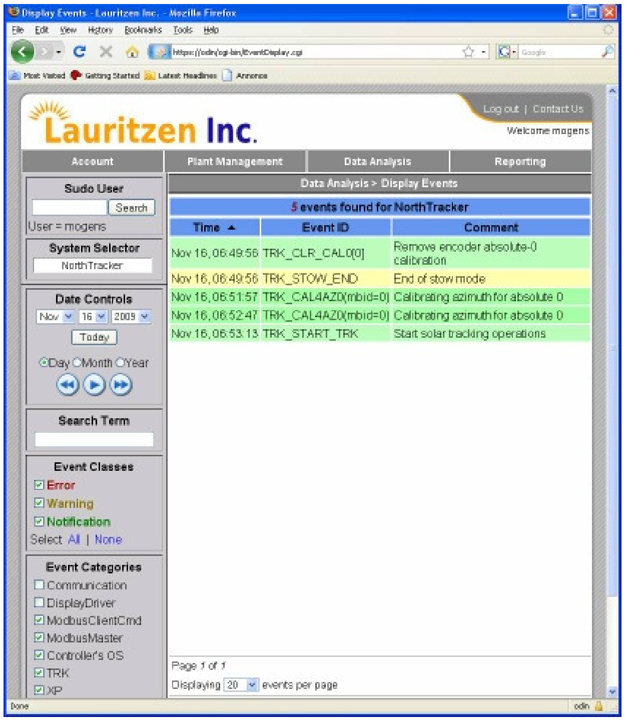
This application note was written to explain the setup and functionality of our Valhalla server, as well as how our controllers are organized within it. We will also go in to detail about how our remote controllers report events to the server, and how alarms can be set from these events.

2. BACKGROUND

All Lauritzen inc. controllers report events to the Valhalla server. Events can be set as alarm triggers, allowing the operator to be notified immediately after events take place. This gives customers the ability to monitor the processes which their systems are going through. The Valhalla server holds detailed historical records for all registered controllers. Valhalla is a multi-threaded Solaris/Linux based server, situated at HQ, which functions as the contact point by remotely based controllers (clients). It uses a MySql database, and provides client data, control and software management.

3. EVENTS

Each controller has a specific set of events that can be generated during its operation. These events provide useful information to the operator, showing the controllers operational items. An example of operational events that an Utgard field controller produces is shown to the left. These events let the user know that the field controller has brought itself out of stow mode, has calibrated the azimuth and is now telling the motor controllers to begin tracking the sun. The Utgard field controller and Idun weather station each have their own set of events that will show the process by which the solar field goes through during operation.


Display Events - Lauritzen Inc. - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Most Visited Getting Started Latest Headlines Annonce
Log out | Contact Us
Account
Plant Management
Data Analysis
Reporting
Search
Today
Done

https://odin/cgi-bin/EventDisplay.cgi

Lauritzen Inc.

Welcome mogens

Sudo User

User = mogens

System Selector: NorthTracker

Date Controls: Nov 16 2009

● Day ○ Month ○ Year

Search Term

Event Classes

☑ Error
☑ Warning
☑ Notification

Select All | None

Event Categories

☐ Communication
☐ DisplayDriver
☑ ModbusClientCmd
☑ ModbusMaster
☑ Controller's OS
☑ TRK
☑ XP

Data Analysis > Display Events

5 events found for NorthTracker

Time ▲	Event ID	Comment
Nov 16, 06:49:56	TRK_CLR_CAL0[0]	Remove encoder absolute-0 calibration
Nov 16, 06:49:56	TRK_STOW_END	End of stow mode
Nov 16, 06:51:57	TRK_CAL4AZ0(mbid=0)	Calibrating azimuth for absolute 0
Nov 16, 06:52:47	TRK_CAL4AZ0(mbid=0)	Calibrating azimuth for absolute 0
Nov 16, 06:53:13	TRK_START_TRK	Start solar tracking operations

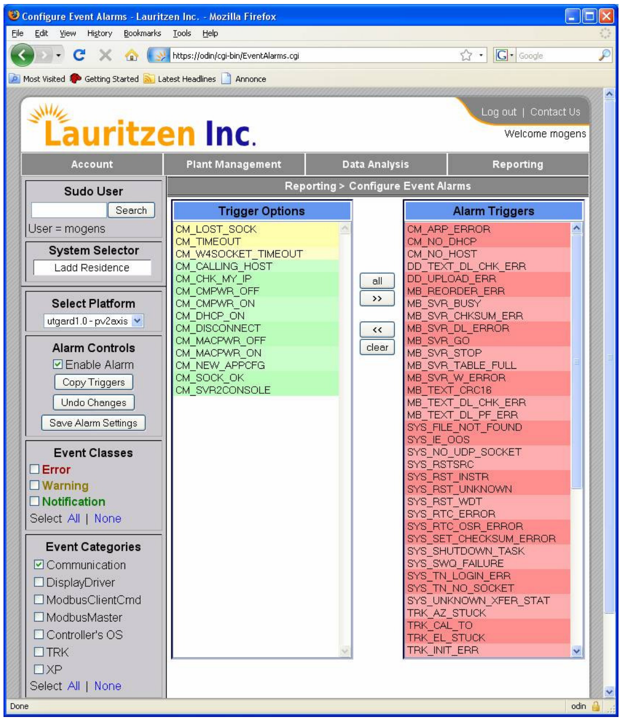
Page 1 of 1

Displaying 20 events per page

odin

4. SETTING ALARMS

One of the most valuable tools that the server provides is the ability to set alarms. This is done by accessing your solar system through Valhalla's web based interface, and setting alarms from a list of possible events. Events are broken down into Categories and Classes to make finding a specific event easier for the user. There are various Categories for each remote controller, ranging from Communication through Operating System to Modbus-master. Classes are grouped as Errors, Warnings or Notifications. Looking at the image to the right, the web page has two separate columns; one for "Trigger Options" and the other for "Alarm Triggers". The user can choose which trigger options they would like to view by selecting the check boxes under "Event Classes" and/or "Event Categories" from the menu of the left. Once the triggers are shown, the user will select which listed trigger options they desire require an alarm and move it to the alarm trigger column. Thus, if a given event is received by the server, and can be matched against an item in the Alarm Trigger list, it will become an alarm. Once an alarm has been triggered, a notification will be sent.



Sudo User

User = mogens

System Selector

Ladd Residence

Select Platform

utgard1.0 - pv2axis

Alarm Controls

☑ Enable Alarm

Event Classes

□ Error
□ Warning
□ Notification
Select All | None

Event Categories

☑ Communication
□ DisplayDriver
□ ModbusClientCmd
□ ModbusMaster
□ Controller's OS
□ TRK
□ XP
Select All | None

Trigger Options

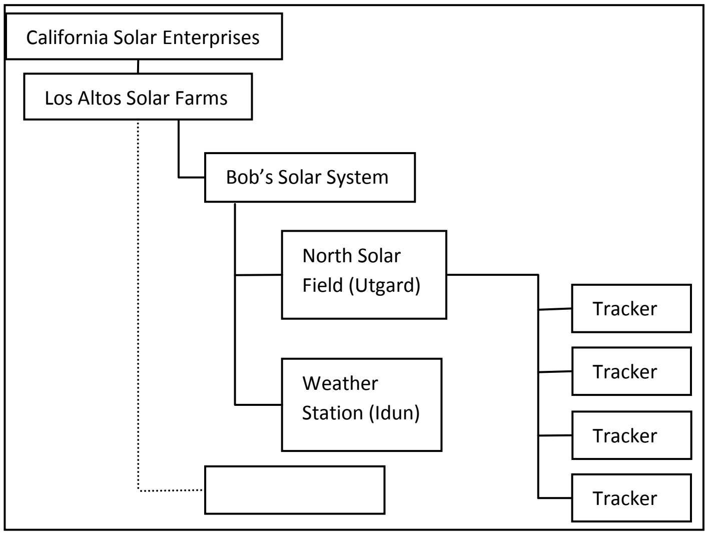
CM_LOST_SOCK
CM_TIMEOUT
CM_W4SOCKET_TIMEOUT
CM_CALLING_HOST
CM_CHK_MY_IP
CM_CMPWR_OFF
CM_CMPWR_ON
CM_DHCP_ON
CM_DISCONNECT
CM_MACPWR_OFF
CM_MACPWR_ON
CM_NEW_APPCFG
CM_SOCK_OK
CM_SVR2CONSOLE

Alarm Triggers

CM_ARP_ERROR
CM_NO_DHCP
CM_NO_HOST
DD_TEXT_DL_CHK_ERR
DD_UPLOAD_ERR
MB_REORDER_ERR
MB_SVR_BUSY
MB_SVR_CHKSUM_ERR
MB_SVR_DL_ERROR
MB_SVR_GO
MB_SVR_STOP
MB_SVR_TABLE_FULL
MB_SVR_W_ERROR
MB_TEXT_CRC16
MB_TEXT_DL_CHK_ERR
MB_TEXT_DL_PF_ERR
SYS_FILE_NOT_FOUND
SYS_IE_OOS
SYS_NO_UDP_SOCKET
SYS_RSTSRC
SYS_RST_INSTR
SYS_RST_UNKNOWN
SYS_RST_WDT
SYS_RTC_ERROR
SYS_RTC_OSR_ERROR
SYS_SET_CHECKSUM_ERROR
SYS_SHUTDOWN_TASK
SYS_SWQ_FAILURE
SYS_TN_LOGIN_ERR
SYS_TN_NO_SOCKET
SYS_UNKNOWN_XFER_STAT
TRK_AZ_STUCK
TRK_CAL_TO
TRK_EL_STUCK
TRK_INIT_ERR

5. SYSTEM HEIRARCHY

In the Valhalla server, customers and their solar systems are placed in to a hierarchy, similar to what is shown in the image below. This hierarchy starts as an Enterprise entry (California Solar Enterprises), moves down to a location of Farms (Los Altos Solar Farms), goes to a specific System level (Bob's Solar System), and finally ends up talking with a field controller(s) (North Solar Field). This hierarchy is important because it allows for events and alarms to be viewed and set at different levels in the chain of command. Both the Utgard controller (or any other remote controller) and Idun weather station report events to the Valhalla server. Once the server receives events, it will start by searching for Alarm Triggers that have been set at the bottom level in the hierarchy (North Solar Field/Weather Station). If no Alarm Triggers are present, the event will move up to the parent level (Bob's Solar Systems), and see if an alarm can been triggered there. The server will continue searching for an alarm until it reaches the top level (California Solar Enterprises), and if no alarms have been triggered it will stop searching. The customer is able to set alarms at any point between the enterprise and field controller levels.



6. ALARM TRIGGERS AND NOTIFICATION

After the customer has gone through the list of possible events and set alarm triggers, the system will need a contact person who will receive notification of any alarms. Contacts can be set up at any level of the hierarchy. A search for a contact person will start at the level at which the alarm was detected. If no contact information is found, the search will move up the hierarchy until a contact can be found and notified. Ultimately, if a contact cannot be found, the alarm is dispatched to one of our System Administrators.

7. APPENDIX (EVENTS)

```
EVENTCM_ARP_ERROR,                  // $e Unable to resolve ARP
EVENTCM_CALLING_HOST,                 // $n Calling host...
EVENTCM_CHK_MY_IP,                  // $n Check that DHCP is bound
EVENTCM_CMPWR_OFF,                   // $n Comm power turned off
EVENTCM_CMPWR_ON,                    // $n Comm power is on
EVENTCM_DHCP_ON,                   // $n DHCP is enabled (displayed on boot)
EVENTCM_DISCONNECT,                  // $n Disconnect
EVENTCM_LOST_SOCK,                  // $n Lost socket connection
EVENTCM_MACPWR_OFF,                   // $n MAC power turned off
EVENTCM_MACPWR_ON,                   // $n MAC power turned on
EVENTCM_NEW_APPCFG,                  // $n Using new AppConfig struct
EVENTCM_NO_DHCP,                   // $e Unable to acquire DHCP issued IP
EVENTCM_NO_HOST,                   // $e CommReboot4DeadHost timeout, unable to contact host
EVENTCM_SOCK_OK,                   // $n Socket connection to host established
EVENTCM_SVR2CONSOLE,                  // $n Server admin connected to client console
EVENTCM_TIMEOUT,                   // $n Host stopped responding...
EVENTCM_W4SOCKET_TIMEOUT,             // $n Timed out in waiting for host connection
EVENTMBCC_MBERR,                   // $w Modbus client command error
EVENTMB_CLEAR_END,                  // $n Done clearing and adding
EVENTMB_CLIENT_RST,                 // $n A client requested a reset sequence
EVENTMB_CLR_MBID,                  // $w All MB id's are being cleared and registry rebuild
EVENTMB_CPWR,                   // $n Modbus controller power changed state
EVENTMB_CPWR_RDY,                   // $n MB has met minimum power turn-on time
EVENTMB_LOCKED,                   // $e MB registry is locked and cannot modified
EVENTMB_REORDER,                   // $n Swapping these two serevr id's
EVENTMB_REORDER_END,                  // $n Done reordering
EVENTMB_REORDER_ERR,                  // $e Attempt to reorder same Sif twice
EVENTMB_REORDER_START,                 // $n Start re-ordering registered Sif's
EVENTMB_SVR_BUSY,                  // $e A MB server is not responding
EVENTMB_SVR_CHKSUM_ERR,               // $e A server found a checksum error during text update
EVENTMB_SVR_DL_ERROR,                 // $e A MB server found a down-load error
EVENTMB_SVR_GO,                   // $e A MB server released it's StopButton
EVENTMB_SVR_OFFLINE,                 // $n A MB server is not communicating
EVENTMB_SVR_ONLINE,                  // $n A MB server started communicating
EVENTMB_SVR_STOP,                  // $e A MB server engaged it's StopButton
EVENTMB_SVR_TABLE_FULL,              // $e Supervisor's modbus table is full
EVENTMB_SVR_W_ERROR,                 // $e A MB server is reporting an error
EVENTMB_TEXT_CRC16,                 // $e Send command to calculate crc16 text checksum
EVENTMB_TEXT_DL_CHK_ERR,              // $e We tried to download new text, but found a checksum error
EVENTMB_TEXT_DL_PF_ERR,              // $e We tried to fownload new text, but received wrong text packet
EVENTMB_TEXT_GET,                  // $n A request to update MB servers was received
EVENTMB_TEXT_LOAD,                  // $n We issued command for MB-servers to load text
EVENTMB_UCLIENT,                   // $n Undefined client, a discovered client was not found
EVENTMB_UPDATE,                   // $n Look-for and register unidentified servers
EVENTSYS_FILE_NOT_FOUND,             // $e File not found on server
EVENTSYS_FSLOAD,                   // $n Request to load file
EVENTSYS_FSUPLOAD,                  // $n Request to upload file
EVENTSYS_FS_LOCKED,                 // $e Cannot create file, FS is locked
EVENTSYS_IE_OOS,                  // $e Out of TCP sockets
EVENTSYS_MPWR,                   // $w $f"(%d)" New Motor Power State
EVENTSYS_NO_UDP_SOCKET,              // $e Idun UDP received unable to secure socket
EVENTSYS_PB_REBOOT,                 // $w System was rebooted from PushButtons
EVENTSYS_RSTSRC,                  // $e Cause for system reset
EVENTSYS_RST_BOR,                  // $w Detected Brown-Out-Reset
EVENTSYS_RST_INSTR,                 // $e Detected Reset caused by reset-instr
EVENTSYS_RST_PD,                  // $n Detected Power-Down (sleep) reset
EVENTSYS_RST_POR,                  // $n Detected Power-On-Reset
```

```
EVENTSYS_RST_UNKNOWN,              // $e $f"rreg=%02x, RunTask=%d" Reset was caused by unknown reasons..
EVENTSYS_RST_WDT,                  // $e $f"(last_task_id=%d)" Detected Watch-Dog-Timer reset
EVENTSYS_RTC_ERROR,                // $e RTC encountered an I2C i/o error
EVENTSYS_RTC_OSR_ERROR,            // $e RTC OscillatorStopFlag detected
EVENTSYS_SET_CHECKSUM_ERROR,       // $e Attempt to reprogram checksum
EVENTSYS_SET_SERNO_ERROR,          // $n Set system serial no
EVENTSYS_SHUTDOWN,                 // $n Shutdown system
EVENTSYS_SHUTDOWN_TASK,            // $e $f" (id=%d)" Task which caused system shutdown and reset
EVENTSYS_STOP,                     // $w $f"(%d)" Stop button state changed
EVENTSYS_SVC_MODE_OFF,             // $w Service mode has been turned off
EVENTSYS_SVC_MODE_ON,              // $w Service mode has been turned on
EVENTSYS_SWQ_FAILURE,              // $e Timeout in getting SWQ signal
EVENTSYS_SW_RST,                   // $n host is requesting a SW reset
EVENTSYS_TN_ADMIN,                 // $n Telnet login as admin
EVENTSYS_TN_GUEST,                 // $w Telnet login as guest
EVENTSYS_TN_LOGIN_ERR,             // $e Illegal telnet login attempt
EVENTSYS_TN_NO_SOCKET,             // $e telnet socket not available
EVENTSYS_UNKNOWN_XFER_STAT,        // $e File xfer protocol error
EVENTTRK_AZ_STUCK,                 // $e $f"(mbid=%d)" Azimuth motor is stuck
EVENTTRK_CAL4AZ0,                  // $n $f"(mbid=%d)" Calibrating azimuth encoder for absolute-0
EVENTTRK_CAL4EL0,                  // $n $f"(mbid=%d)" Calibrating elevation encoder for absolute-0
EVENTTRK_CLEAN_DET,                // $n Start of clean mode
EVENTTRK_CLEAN_END,                // $n End of clean mode
EVENTTRK_CLEAN_POS,                // $n Trackers are in clean mode
EVENTTRK_CLR_CAL0,                 // $n Remove encoder absolute-0 calibration
EVENTTRK_EL_STUCK,                 // $e $f"(mbid=%d)" Elevation motor is stuck
EVENTTRK_INIT_ERR,                 // $e Modbus is taking too long to come online
EVENTTRK_MOVE_TO,                  // $e Timeout occurred during move process
EVENTTRK_NO_IDUN,                  // $e Idun is not broadcasting weather info
EVENTTRK_START_TRK,                // $n Start solar tracking operations
EVENTTRK_STORM_DET,                // $w Detected storm condition
EVENTTRK_STORM_END,                // $w End of storm condition
EVENTTRK_STORM_POS,                // $w Trackers are in storm position
EVENTTRK_STOW_DET,                 // $n Start of stow mode
EVENTTRK_STOW_END,                 // $n End of stow mode
EVENTTRK_STOW_POS,                 // $n Trackers are in stow mode
EVENTXP_MB_ERROR,                  // $w $f"(mb_stat=%02x)" Transporter saw Modbus read error
EVENTXP_UNEXP_TGT,                 // $e Unexpected target addr in reply
```