

# Logger Configuration Management Software - Functional Specification

## Overview

This document is a companion to the document *Logger Firmware Functional Specification*. Whereas that document describes the operation of the logger itself, this one describes the function of software that operates on a server machine, for the purpose of managing the operation of one or more logger devices.

The reader is referred to that document to gain a more complete picture of the system, and in particular for specification of the XML strings that are exchanged between Configuration Management Software and the Logger.

The Configuration Management software (sometimes referred to as "herd control" software) has the following characteristics:

- Actually three software modules
  - Logger Interaction
    - Stores all configuration information about some number of logger devices in an SQL database (the actual choice of database product is not important, but the initial version uses MySQL).
    - Parses XML strings sent from the logger, compares this information, as appropriate, with information stored in the database.
    - Sends an XML string to the logger, containing configuration updates, as appropriate.
    - The XML is as defined in the *Logger Firmware Functional Specification*
  - User Interface
    - User Interface software module that presents logger information in tabular form, gives entry boxes for configuration that needs to be changed..
  - Status Monitor
    - Monitor logger pool for incorrect operation, notify of problems

Configuration management is described here as if it is independent of the management of reported data. In reality, whatever server is used to process data needs to know some of the information that is stored in the configuration database. In the long term, it is entirely possible that these functions could be merged. For the sake of initial development (and particularly debug), however, it is believed that it will be advantageous to treat these as separate functions.

## ***Database Interaction with Logger***

The central element of the logger configuration management is a database that holds information about the attached devices. The various data fields, and their relationship to the XML strings exchanged with the logger, are described below.

- There is a table containing a record for each logger. This record contains the basic configuration, plus some status information about the logger:
  - ID – the unique ID of the logger
    - XML Fields
      - `<LOGGER_UPLOAD><CONFIG><LOGGER><ID>` (upload)
      - reported by the logger, and used to identify relevant records in the database
      - If an unrecognized ID is received, a new database record is created, assuming that a new logger is registering itself.
  - CFGURL – the URL to which logger configuration information will be reported.
    - Reported by the logger
    - XML Fields
      - `<LOGGER_UPLOAD><CONFIG><LOGGER><CFGURL>` (upload)
      - `<LOGGER_DOWNLOAD><LOGGER><CFGURL>` (download)
    - If the database contains no entry, the value reported by the logger is stored in the database.
    - Otherwise, if the database entry is different from the reported value, the database entry is sent back to the logger, with the assumption that the logger will update its value and subsequently use the new one.
  - LOGURL – the URL to which logged data will be reported.
    - Reported by the logger
    - XML Fields
      - `<LOGGER_UPLOAD><CONFIG><LOGGER><LOGURL>` (upload)
      - `<LOGGER_DOWNLOAD><LOGGER><LOGURL>` (download)
    - If the database contains no entry, the value reported by the logger is stored in the database.
    - Otherwise, if the database entry is different from the reported value, the database entry is sent back to the logger, with the assumption that the logger will update its value and subsequently use the new one.
- CRON – a string, in standard "cron" format, that specifies the times at which

collected data will be reported.

- Reported by the logger
- XML Fields
  - <LOGGER\_UPLOAD><CONFIG><LOGGER><CRON> (upload)
  - <LOGGER\_DOWNLOAD><LOGGER><CRON> (download)
- If the database contains no entry, the value reported by the logger is stored in the database.
- Otherwise, if the database entry is different from the reported value, the database entry is sent back to the logger, with the assumption that the logger will update its value and subsequently use the new one.
- UPTIME – The time, in seconds, since the last logger general reset.
  - Reported by the logger
  - XML Fields
    - <LOGGER\_UPLOAD><CONFIG><LOGGER><UPTIME> (upload)
    - Stored in the database as information useful in determining the need for field service or replacement.
- LOCATION – a text field that exists only in the configuration database, that identifies the location of the logger in human-readable format.
  - Informational only.
  - No XML transfers
- LAST\_REPORT – the time, in seconds since midnight, 1/1/1970, of the last report
  - Generated by software at time of report.
  - Stored in database
  - No XML transfer
- DOWNLOAD – a field that names a new firmware version. A URL.
  - XML Fields
    - <LOGGER\_DOWNLOAD><LOGGER><DOWNLOAD> (download)
    - If not empty, then this field is sent to the logger, which subsequently accesses this URL and downloads the new firmware. Logger subsequently restarts using the new firmware.
- RCSIDS – a string that contains the software version numbers of all of the software modules in the logger.

- XML Fields
  - <LOGGER\_UPLOAD><CONFIG><LOGGER><RCSIDS> (upload)
  - Reported by the logger, stored in database. Useful in system management.
- OSVER – a string representing the operating system level being used by the logger.
  - XML Fields
    - <LOGGER\_UPLOAD><CONFIG><LOGGER><OSVER> (upload)
    - Reported by the logger, stored in database. Useful in system management.
- There is also a table containing a list of readable devices. Each record contains the following information:
  - ID – the unique ID of the device
    - XML Fields
      - <LOGGER\_UPLOAD><CONFIG><RDRS><RDR><ID> (upload)
      - Reported by logger, used to identify relevant record in database.
      - If an unrecognized ID is received, it is assumed that a new device has been received, and a new database record is created.
  - CHAN – the channel within the device
    - XML Fields
      - <LOGGER\_UPLOAD><CONFIG><RDRS><RDR><CH> (upload)
      - LOGGER\_ID – The ID of the logger to which the device is attached
    - Reported by logger, used to identify relevant records in database.
  - CRON – the timing specification for sampling data
    - Reported by the logger
    - XML Fields
      - <LOGGER\_UPLOAD><CONFIG><RDRS><RDR><CRON> (upload)
      - <LOGGER\_DOWNLOAD><RDRS><RDR><CRON> (download)
    - If the database contains no entry, the value reported by the logger is stored in the database.
    - Otherwise, if the database entry is different from the reported value, the database entry is sent back to the logger, with the assumption that the logger will update its value and subsequently use the new one.
  - POLICY – the sampling policy

- Reported by the logger
- XML Fields
  - <LOGGER\_UPLOAD><CONFIG><RDRS><RDR><POL> (upload)
  - <LOGGER\_DOWNLOAD><RDRS><RDR><POL> (download)
- If the database contains no entry, the value reported by the logger is stored in the database.
- Otherwise, if the database entry is different from the reported value, the database entry is sent back to the logger, with the assumption that the logger will update its value and subsequently use the new one.

### ***Other XML***

In addition to the XML described above, the Configuration Management Software returns to the Logger a <LOGGER\_DOWNLOAD><LOGGER><CRON><TIME> XML sequence containing current time. Presumably, the logger uses this value to update its clock.

### ***User Interface***

A second part of the configuration management software is a program that gives a system administrator the ability to view the configuration and status of all loggers, and also to alter their configuration. This piece interacts only with the database, and has the following functions:

- For each logger:
  - A read-only display of the logger's ID
  - A read/write display of the logger's "location"
  - A read-only display of the logger's "Configuration URL"
  - A read/write display of the logger's "Log URL"
  - A read-only display of the logger's OS Level.
  - A read/write display of the logger's "CRON" string.
  - A writeable display of the software update URL.
  - A read-only display of the number of hours since the logger last had a reset
  - A read-only display of the time of last logger contact.
- For each readable device on the logger
  - A read-only display of the device's ID & channel
  - A read/write display of the times at which the device should be read, in "cron"

format.

- A read/write display of the current sampling policy for this device and channel

For all read/write displays, a change causes the database value to be overwritten. In some cases, this causes the logger to be updated at its next report. This is described in the section on database logger interaction.

Although the user interface has the appearance of a "control panel" for programming loggers, it must be remembered that all communications are initiated by the loggers themselves, according to their configuration. There will hence be a time delay, possibly very long, before configuration changes take effect.

### ***Health Monitors***

The function of this software is largely TBD, and will evolve.

It is intended that there be a periodic check, probably executed by a "cron" on the server, that reads the database and checks certain critical aspects of logger operation.

Initially, it will simply look for and report (with an email?) failure of a logger to report at its designated time.

It will also be useful to monitor "up time" to see if a logger is suffering from frequent system resets, an indication of either bad power or a system problem.