COMMUNICATION AND INFORMATION SOLUTIONS FOR A SAFER WORLD

System Development

IDD

Logging

EWSI00EN60112.10

HISTORY CHART

VERSION	DATE	CHANGED CHAPTERS	CAUSE OF CHANGE	AUTHOR
1.0	2013-06-07	All chapters	Log File Format 2, restricted to log file format definition and removed other stuff	A. Reisenbauer

Total number of pages: 13

Latest version approved by: S. Kreuzhuber

Latest version released by: M. Schultschik

FREQUENTIS AG
Innovationsstr. 1, 1100 Vienna, Austria, http://www.frequentis.com/
Vienna Commercial Court, FN 72115 b, ATU 14715600, DVR 0364797

CONTENTS

ILLUSTRATIONS

TABLES

1 Introduction

1.1 Purpose

This document describes the log file format used for logging from any Frequentis Device or Software wherever applicable.

The log file format is not depending on a programming language like C#, JAVA, or C++. It describes the format of the text file. If a custom logging system is developed, this IDD document should be taken in account.

1.2 Target Group

This document is intended to all developers involved in the Software Development activities.

1.3 System Overview

Each application writes some service messages to text files, which can be revised later by system engineers or developers. Typically, there are written messages about fatal problems, errors, exceptions, operational states and user activities. To be able to use common tools for log file viewing, format of the log file is unified in this document.

This document replaces the File Format section of the [IDDLOG] without touching the other specified items of that IDD (file location, file naming convention, PC Dispatcher implementation details).

1.4 References

Document	Document Identifier
[IDDLOG] FREQUENTIS' IDD LOGFILEFORMAT	PS0A20EN640.10
[ISO 8601] Data elements and interchange formats — Information interchange — Representation of dates and times	ISO 8601:2004(E)
UTF-8, a transformation format of ISO 10646 (http://tools.ietf.org/html/rfc3629)	[RFC 3629]
Common Format and MIME Type for Comma-Separated Values (CSV) Files (http://tools.ietf.org/html/rfc4180)	[RFC 4180]
Augmented BNF for Syntax Specifications: ABNF (http://tools.ietf.org/html/rfc5234#appendix-B.1)	[RFC 5234]
The Syslog Protocol (http://tools.ietf.org/html/rfc5424#page-10)	[RFC 5424]

1.5 Definitions

Within this document “**System**” stands for any FREQUENTIS System.

Within this document “**Computer Software Configuration Item**” (**CSCI**) stands for an aggregation of software that satisfies an end use function and is designated for separate configuration management by the acquirer. CSCIs are selected based on trade-offs among software function, size, host or target computers, developer, support concept, plans for reuse, criticality, interface considerations and other factors [MIL-STD-498]. Each CSCI needs to be separately documented and controlled.

2 Interface Design Description

2.1 Interface Overview

Any application writes messages to special text files. With the specific format described in this IDD we call it "log file". Each log line contains typically timestamp, severity level, and a message for the user.

2.2 Log File Format

2.2.1 Overview

Log file is a text file containing printable ASCII characters only and (optionally) UTF-8[1] encoded characters. It consists of log entries. Each log entry is written to one either Internet standard newline[2] or UNIX line ending LF ('0x0A'). One log entry consists of elements delimited by semicolons ('0x3B'). This format allows handle it as CSV file[3].

- First log line contains format description of one log entry (optional).
- Second log line contains logger service information about the log file, logger settings, original placement of the log file, original file name (optional).
- Third log line contains information about application, which writes log entries to the file (optional).
- Following lines contain messages from application (mandatory).
- Last line contains special message informing about closing log file (optional).

[1] According [RFC 3629]

[2] According to [RFC 5234] CRLF = CR ('0x0D') LF ('0x0A')

[3] The format definition given here does not follow the CSV Format as specified in [RFC 4180]. This is mainly for backward compatibility reasons. Nevertheless, any application designed for handling the log format SHOULD be able to handle well-formed CSV according [RFC 4180] also.

2.2.2 First Log Line – Log Entry Format - Optional

The first line contains information about log entry format. Each element of the log entry is describing data type which is available in particular column. This information will be used by log viewer when displaying log entry for user. If the log file contains elements (columns) not known for log viewer, it will skip them, or interpret in some default way, e.g.: as plain text.

First line of the log file – Logfile Format v1:

```
dd.MM.yyyy HH:mm:ss,000; sever; prcId; [title]; message
```

Fig. 1: Format log line in log format version 1

First line of the log file – Logfile Format v2:

```
YYYY-MM-DDTHH:mm:ss,ssssss+HHmm; sever; HostId; ctxtId; [title]; message;
```

Fig. 2: Format log line in log format version 2

Entry	Type	Description	Version
`dd.MM.yyyy HH:mm:ss,000`	Date and Time	Date and time when the log entry was created. Milliseconds part is separated by comma. Leading zeroes are added (all parts) Format of date-time stamp: dd.MM.yyyy HH:mm:ss,000. Example: 14.03.200710:06:55,263	1
`YYYY-MM-DDT HH:mm:ss,ssssss +HHmm`	Date and Time[4]	Date and time when the log entry was created. Microsecond's part is separated by comma. Leading zeroes are added (all parts) Example: 2007-03-14T10:06:55,263456+0200	2
severity	String n chars	Log entry severity. Following severities will be supported: • FATAL • ALERT (new in v2) • CRITICAL (new in v2) • ERROR • WARN • NOTICE (new in v2) • INFO • DEBUG • TRACE (new in v2)[5] This list of severities might be extended in the future.	1 & 2
HostId	String n chars	ID of the host generating the log entry. String must not contain any ";" character. Examples for Host IDs are: IPv4 Address, IPv6 Address,	2

[4] According to [ISO 8601]

[5] Even if it is NOT recommended for using the logging system for tracing purpose

		FQDN, Hostname, any unique ID	
prcId	‘P’ int 4 digits	Process ID which made log entry starting with P character. Example: P2496 The length of this field is minimum 4 characters, leading ‘0’ are added where necessary.	1
ctxtId	‘P’|’T’ 4 digits or n chars	Context ID (which can be a Process ID – as in version 1 or a Thread ID, or something else) which made log entry starting with P (rocess) or T (hread) character. Example: P2496 The length of this field is minimum 4 characters, leading ‘0’ shall be added in case of numeric ctxtId.	2
title	[string]	Identification of the log entry source closed in square brackets. It can be name of the source file, line number, module or process name. Examples: [S/UserManage.CopyAndValidate] [StateMachine.c, line 255] [FRQ-NmsElmasServer.exe] Recommendation: Keep a constant length per application to have it consistent and more readable, truncate the string or pad with blanks where necessary.	1 & 2
message	string	message for user, the semicolon ‘;’ characters will be replaced by comma ‘,’.[6] If multiple EOLs (End Of Lines) are inside the message, they shall be split to several log lines. Each log line shall contain header (from date to title) copied from leading log line.	1
message	string	message for user, Multiple EOLs (End Of Lines) inside the message without any quoting[6] are allowed for better readability. This field may or may not be enclosed in double quotes. If the field is not enclosed with double quotes (“”), the semicolon ‘;’ character may not appear inside the message. If double-quotes are used to enclose this field, a double-quote appearing within the message must be escaped by preceding it with another double quote (“” → “”””).[7]	2

Tab. 1: Log entry format.

[6] The format definition given here does not follow the CSV Format as specified in [RFC 4180]. Nevertheless, any application designed for handling the log format SHOULD be able to handle well-formed CSV according [RFC 4180] also.

[7] Especially at embedded devices the application using the logging shall be aware of performance penalties introduced through escaping and replacing characters. Therefore, if ever possible these characters should be prevented within the log string whenever possible.

Implementation hint: The time information shall be formatted by logger subsystem on the client side. Local time or UTC time will be used depending on configuration.

The PC Dispatcher is currently using following format for the ‘title’ element:

```
[<PI>/<Class>.<Method>]
```

- PI – is program identifier S=PCD server; C=PCD client
- Class is name of the class
- Method is name of the method

Hints for the utilisation of severity levels[8]

For greatest possible standardisation of log entries utilisation of severity classes shall follow:

FATAL:	System is unusable
ALERT:	Action must be taken immediately
CRITICAL:	Critical conditions
ERROR:	Error conditions
WARN:	Warning conditions
NOTICE:	Normal but significant condition
INFO:	Informational messages
DEBUG:	Debug-level messages
TRACE:	Tracing purpose

2.2.3 Second Log Line – Service Information from Logger - Optional

This log entry will contain in the ‘message’ element following information:

- Path and log file name
- Maximum log file size
- Maximum total size of all log files
- Number of days log file will be kept in the directory, then will be deleted

The log entry might contain also other information depending on transport mechanism type and logger service implementation.

Example - version 1:

```
25.07.2006 10:18:15,296; INFO;  P1234; [FRQ-LoggerService]; Opened new file
C:\Program Files\Frequentis\LOG\ELMAS\20060725-101815.LOG SECTION=ELMAS
MAXLOGSIZE=1000000 MAXTOTALSIZE=500000000 DAYSTOKEEPLOGFILE=30
```

Fig. 3: Logger information log line in log format version 1

2.2.4 Third Log Line – Service Information from Application - Optional

This log entry will contain service information from application. It might not be used by application, it is optional.

[8] Following the idea of “syslog” standardisation in [RFC 5424] – http://tools.ietf.org/html/rfc5424#page-11

E.g., PC Dispatcher logs following information:

- PCDispatcher version
- Hostname of the working position
- COI version
- Logged-in User
- Logged-in Role
- CCI-Port (if TETRA is used)
- Interface address of selected and monitoring ERIF
- Version of TAL

Example - version 1:

```
24.01.2007 16:19:25,405; INFO;  P3116; [S/CommonData. SetLogHeader]; PCDispatcher
version:3.5.61.9000 Hostname:WP002 COI version:1.3.3.15 User:OP2 Role:Operator CCI
port:WP002 SEL erif:09:0:1202 MON erif:09:0:1203 TAL version:v1.1.4322
```

Fig. 4: Application information log line in log format version 1

This line is optional and no viewer shall rely on its presence in the log file.

2.2.5 Following Log Lines – Messages from Application

Following log entries will contain messages from application.

2.2.5.1 Version 1

If the log message contains EOL (CR, LF) then it will be split to multiple lines, as they was logged by several calls. Each line will be filled with standard header copied from first line.

Example of standard log line:

```
05.12.2006 13:31:06,950; DEBUG; P1088; [S/InterfaceM. AddInterface]; Add interface
(Interface: 00:0:7777)

05.12.2006 13:32:44;501; ERROR; P2624; [Frq-NmsElmasServer]; Save data failure.
```

Fig. 5: Standard log line in log format version 1

2.2.5.2 Version 2

With log format version 2, multiline log entries without quoting were introduced.

Example of standard log line

```
2006-12-05T13:31:06,950459+0200; DEBUG; hansi.frequentis.frq; P1088; [S/InterfaceM.
AddInterface]; Add interface (Interface: 00:0:7777)

2006-12-05T13:32:44,501123+0200; ERROR; 10.14.12.234; P2624; [Frq-NmsElmasServer];
Save data failure.
```

Fig. 6: Standard log line in log format version 2

2.2.6 Last Log Line – Closing File Message - Optional

The last line will be filled by logger service with information about closing file.

2.3 Naming of Log Files - Optional

As this IDD is focused on the format of the log information rather than the naming convention of log files, this section has to be seen as a recommendation only.

The file name should contain the time of log file creation and an identification, which is really helpful in the given context. For example:

```
"yyyyMMdd_HHmmss_"+<someidentification>+".log"
```

but could also be vice versa.

UTC/ local time will be used depending on configuration.

3 Abbreviations

ASCII	American Standard Code for Information Interchange
CR	Carriage Return
CSV	Comma (or Character) Separated Values
EOL	End Of Line
FRQ	Frequentis
ID	Identification
IDD	Interface Design Description
LF	Line Feed
Rev.	Revision
UCS	Universal Character Set
UTC	Coordinated Universal Time
UTF-8	UCS Transformation Format-8Bit