NASA GSFC FLIGHT SOFTWARE SYSTEMS BRANCH

FSW VERSION DESCRIPTION DOCUMENT

CFS HEALTH & SAFETY (HS) APPLICATION

BUILD: HS 2.4.0

RELEASE DATE: 8/27/2021

1.0 FSW VERSION DESCRIPTION

1.1 PURPOSE AND SUMMARY

This build is a minor build of the Health & Safety (HS) application to resolve bugs with the HS application. This build does not include any new functionality. The primary purpose of this release is to ensure compatibility between the HS application and cFS Caelum.

This document serves as the notification of the Build 2.4.0 release of the cFS HS application.

1.2 NEW/CHANGED FUNCTIONALITY IN THIS VERSION

Table 1.2-1 identifies the DCRs that have been implemented in this FSW version. For each DCR the "Key" column shows the corresponding DCR in the GSFC cFS tracking system.

Table 1.2-1 – DCRs Implemented in this Version

Key	Summary	Description
GSFCCFS-1089	Migrate HS unit tests to distributed UT Assert	
GSFCCFS-1172	HS does not appear to handle short format of events	This was an issue exposed when addressing ticket GSFCCFS-1151 (compatibility with cFE 6.8). For event monitoring, HS previously relied on long format events (subscribing to CFE_EVS_EVENT_MSG_MID in HS_AppMain and handling CFE_EVS_Packet_t in HS_MonitorEvent). In cFE 6.8, these no longer exist, and in fact were separated to handle long format and short format event messages differently. The backwards compatibility defines in cFE map these things to the event long formats. typedef CFE_EVS_LongEventTlm_t CFE_EVS_Packet_t; #define CFE_EVS_EVENT_MSG_MID CFE_EVS_LONG_EVENT_MSG_MID For purposes of resolving GSFCCFS-1151, occurrences of the deprecated items were fixed with the long format, but in the long term, HS needs updates to handle monitoring short format event messages.
GSFCCFS-1182	HS has static code analysis findings	In analysis done on 7/10/2020, CodeSonar flagged the attached findings.

GSFCCFS-1259	Update HS to use new cFE Message Module	
GSFCCFS-1479	HS does not build with eval-cert3	
GSFCCFS-1581	HS doxygen config file should be renamed for clarity	The filename "hs_config.txt" suggests that this a configuration file for the app itself as opposed to a configuration file for doxygen.
GSFCCFS-1589	HS should use const for function arguments where possible	
GSFCCFS-1598	HS missing requirement for MAT loading on initialization	HS loads 4 tables: the Execution Counter Table (XCT), the Application Monitor Table (AMT), the Events Monitor Table (EMT), and the Message Actions Table (MAT). Three of those tables (XCT, AMT, and EMT) have requirements specifying that if the table fails to load during initialization, the corresponding monitoring will be disabled and an event message will be sent. The MAT has the same behavior, but no associated requirement. The proposed action to add 2 new requirements for HS matching the format of 8003 and 8003.1, but relating to the MAT.
GSFCCFS-1599	Proposed HS8007: Upon any Initialization, HS shall load the Message Actions Table.	Upon any Initialization, HS shall load the Message Actions Table.
GSFCCFS-1600	Proposed HS8007.1	Description: If the Message Actions Table fails validation, HS shall issue an event message and disable the use of the Message Actions Table. Rationale: Can verify gross errors in table. If errors exist, HS can still perform other monitoring (just cannot use message actions).
GSFCCFS-1623	HS event messages do not allow for extended message IDs	Events that print out a messageID value use the 0x04X format specifier, which does not work for longer message IDs.
GSFCCFS-1713	HS should use strict resource IDs	

No new functionality was added in this build.

1.3 MISSING PLANNED FEATURES AND KNOWN PROBLEMS

Table 1.3-1 identifies currently open DCRs that are not addressed in this build.

Any workarounds that may apply are identified.

Table 1.3-1 – Currently open DCRs

Key	Summary	Description
GSFCCFS-1014	Scrub configuration file for items that don't change	Suggest removing non-project configuration items from this file. Would help to limit this list to only items that projects should expect to manage. Examples of things I wouldn't consider for project configuration - app name, wakeup pipe depth, table names, etc. Note open ticket on CPU utilization/idle task so avoiding comments on those parameters since I expect this implementation to change.
GSFCCFS-1208	Performance ID usage is inconsistent in HS app	Anytime a task goes into a system delay (for example a pend on message receipt or task delay), an app should do the following: CFE_ES_PerfLogExit(<performance_id>); <do the delay function> CFE_ES_PerfLogEntry(<performance_id>); There appear to be some cases in HS where a task delay called, but the performance monitor is not exited.
GSFCCFS-1003	Recommended refactoring in HS_MonitorEvent	In HS_MonitorEvent, Jake recommends: refactor common action logic for the different monitors into one function, and pass in unique info. Would avoid repeated logic for the same action from a different trigger. finding from JSC code review
GSFCCFS-727	HS idle task consumes 100% CPU on POSIX systems	
GSFCCFS-726	HS idle task needs modification to support SMP	Idle task needs to be updated for SMP so that utilization can be determined for each core.

2.0 DELIVERED PRODUCTS

Table 2-1 identifies the locations of FSW products relevant to this FSW Build. The version or date of the Build and where the product can be located are provided. Changes from a previous VDD are identified.

Table 2-1 – Delivered Products and their Locations

Software Element	Changed with this Version?	New Version or Date	Location
Source Code of this FSW Build	Yes	2.4.0	https://github.com/nasa/hs
Doxygen Documentation	Yes	N/A	https://github.com/nasa/hs
Unit Test Data	Yes	2.4.0	https://github.com/nasa/hs
FSW Make Files	Yes	2.4.0	https://github.com/nasa/hs

3.0 INSTALLATION PROCEDURES

In order to build and install the HS application, it must be added to the cFE CMake build system. This is done by modifying the TGTX_APPLIST in the cFE targets.cmake file. This is shown in the trivial example below.

```
SET(TGT1_NAME cpu1)
SET(TGT1_APPLIST hs)
SET(TGT1_FILELIST cfe_es_startup.scr)
```

After HS is added to the targets.cmake file, it is built and installed using the standard cFE CMake build instructions. These instructions are available in cFE CMake documentation:

https://github.com/nasa/cFE/blob/main/cmake/README.md

4.0 CONFIGURATION SUMMARY AND VERSION IDENTIFICATION

This software can be found in the HS GitHub repository (https://github.com/nasa/HS) under the tag "2.4.0".

Verification of the version can be done by sending an HS NOOP command which produces an event message containing the version information. In addition, the initialization event message generated during the application startup provides the version information.

ACRONYMS

ACS ..... Attitude Control System
C&DH..... Command and Data Handling
cFS..... Core Flight System
CM ..... Configuration Management
COTS ..... Commercial Off-The-Shelf
CPU ..... Central Processing Unit
DCR ..... Discrepancy/Change Request
ETU..... Engineering Test Unit
FSB..... Flight Software Branch
FSW..... Flight Software
GSFC..... Goddard Space Flight Center
HS..... Health & Safety
I&T..... Integration & Test
JSC ..... Johnson Space Center
POSIX..... Portable Operating System Interface
RTOS..... Real-Time Operating System
SMP ..... Symmetric Multiprocessing
T&C..... Telemetry and Command
TBD..... To Be Determined
URL..... Universal Resource Locator
VDD ..... Version Description Document