



F', Open Source, and the Future

Michael Starch
NASA Jet Propulsion Laboratory

Copyright © 2023 California Institute of Technology.
Government sponsorship acknowledged.



Jet Propulsion Laboratory
California Institute of Technology

Agenda

- F' and Open Source
- Benefits of Open Source
- Lessons Learned
- F' and the Future

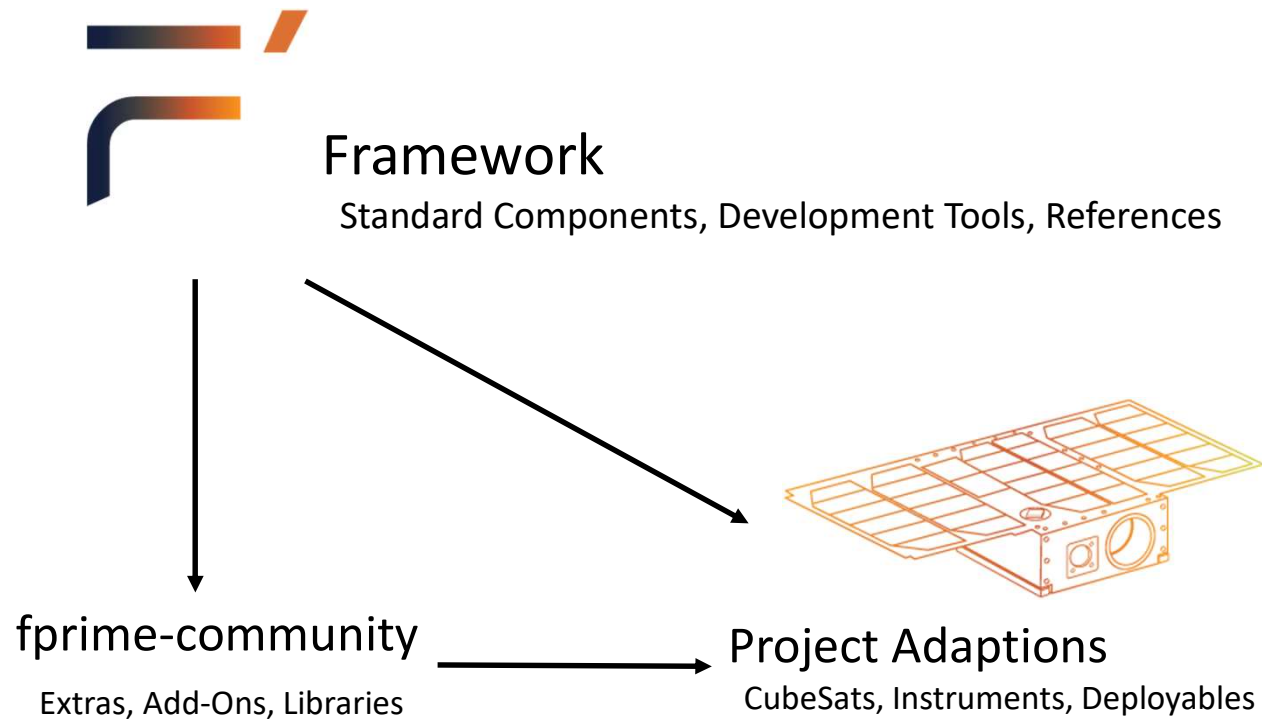
Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.



F' and Open Source



Framework and Projects



Open Sourcing



Jet Propulsion Laboratory
California Institute of Technology



<https://octodex.github.com/original/>

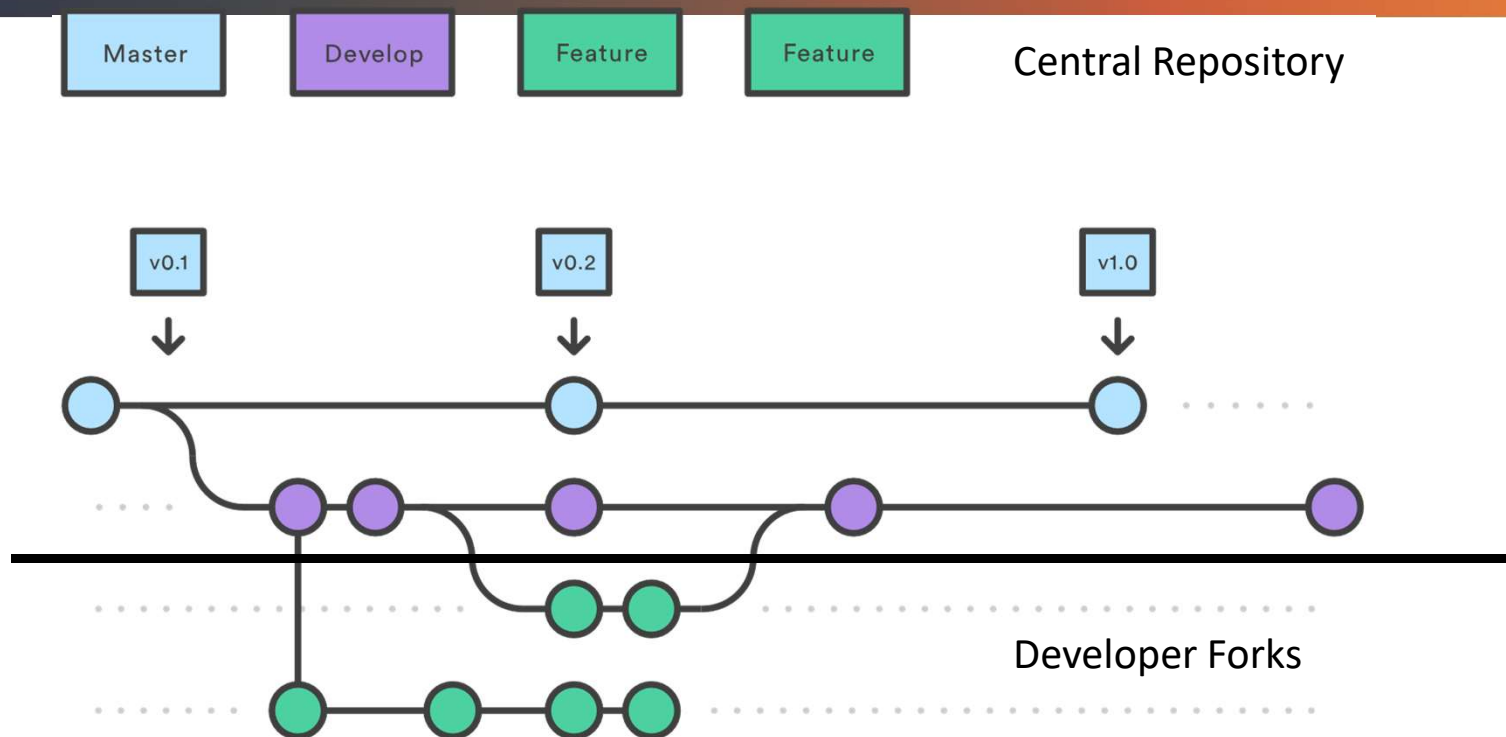


2013	Internal Development	2017	Open Development	2023
------	----------------------	------	------------------	------

Apache License, Version 2.0



F ' Open Development Model



<https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>



Open Tooling and Documentation



Static Analysis







<https://www.doxygen.nl/index.html>



<https://www.sphinx-doc.org/en/master/>

Generated Documentation

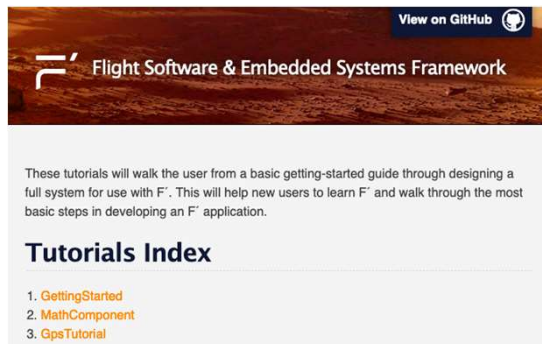
✓	All checks have passed 2 successful and 2 neutral checks	Hide all checks
✓	 CI / build (pull_request) Successful in 20m	Required Details
■	 LGTM analysis: JavaScript Completed in 42s — No code changes detected	Details
■	 LGTM analysis: Python Completed in 42s — No code changes detected	Details
✓	 LGTM analysis: C/C++ Successful in 7m — 1 new alert	Details

Automatic Pull Request Verification

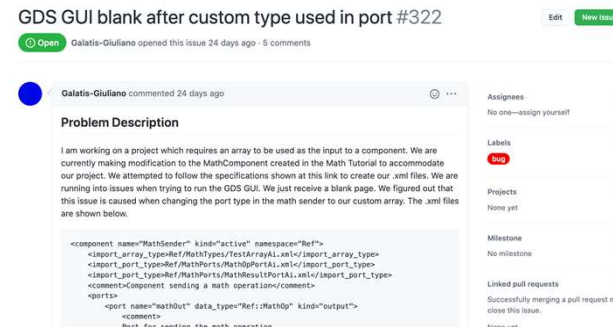


Benefits of Open Source

Extended Oversight



Documentation Edits



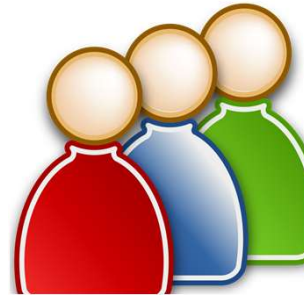
Error Reports



Design and Code Reviews



Community of Expertise



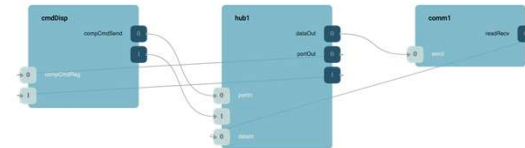
<https://commons.wikimedia.org/wiki/File:User-group-square.svg>



Best Practices



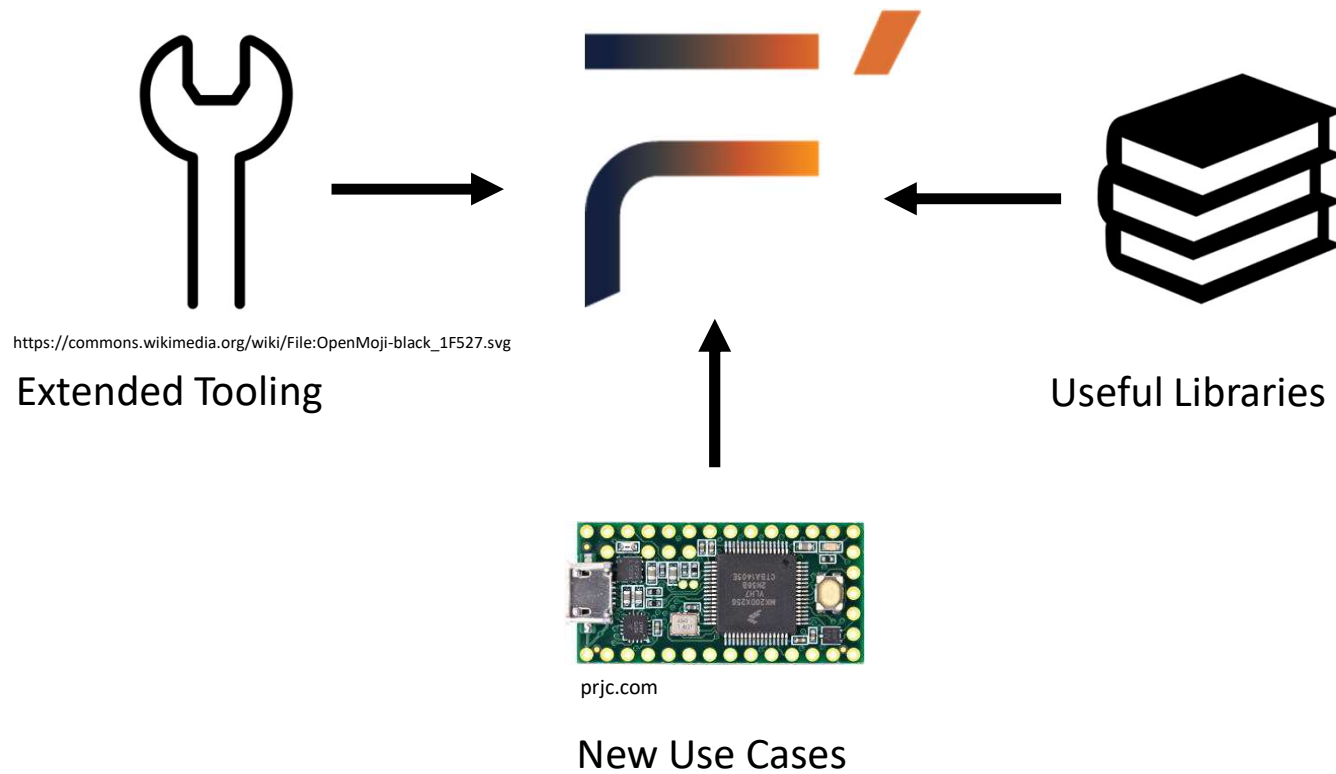
Answers



Recommended Patterns



Ecosystem Surrounding Framework

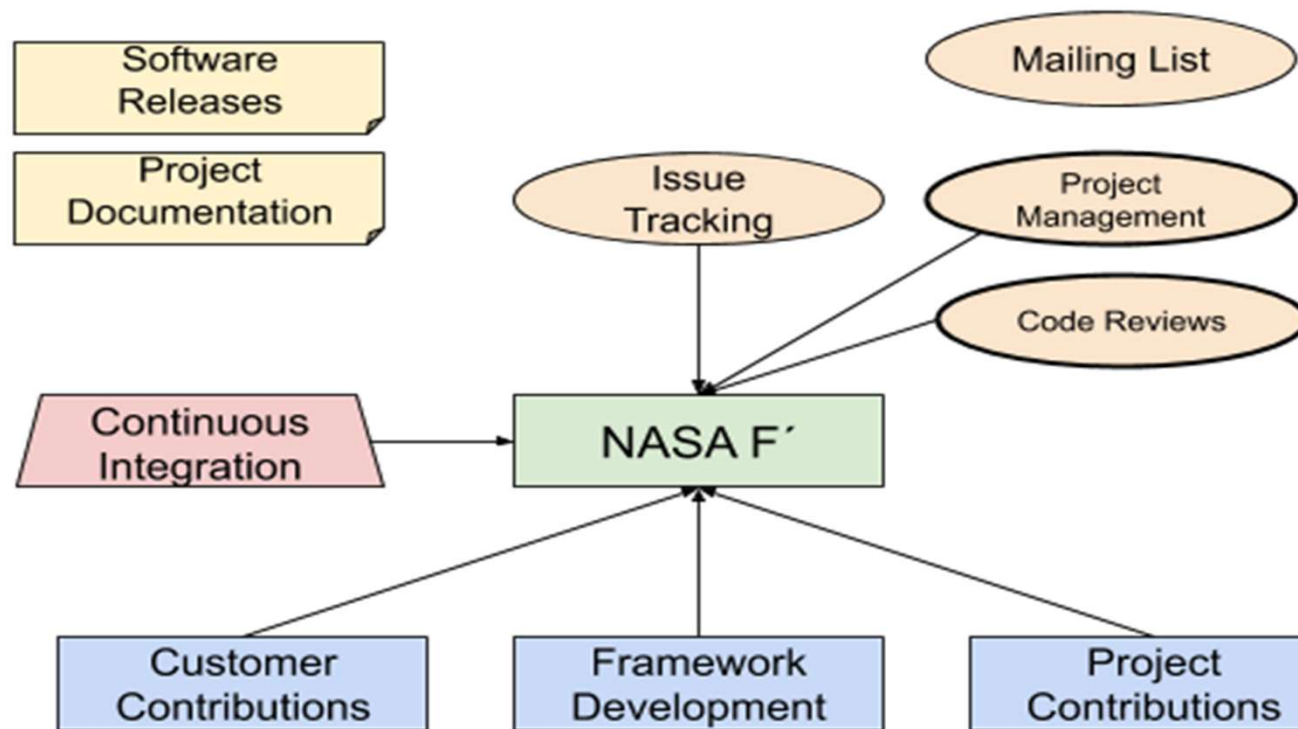




Lessons Learned

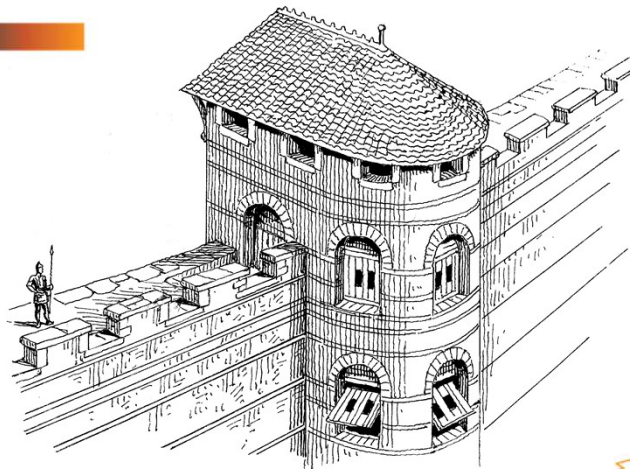


Embrace Openness

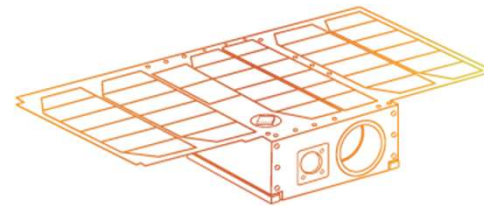




Separate Product Line from Projects



Pearson Scott Foresman, Public Domain



Budget, Schedules, Short-Cuts



Leverage Tools and Automation



Free and Open



<https://docker.com>

Ease Maintenance



Standard Practice



Low Integration Overhead



<https://github.com/cookiecutter/cookiecutter>

Reduce Development



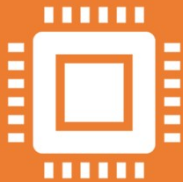
F' and The Future



The F' Product Line Roadmap



Maintenance



Baremetal
Support



Class-B
Standards























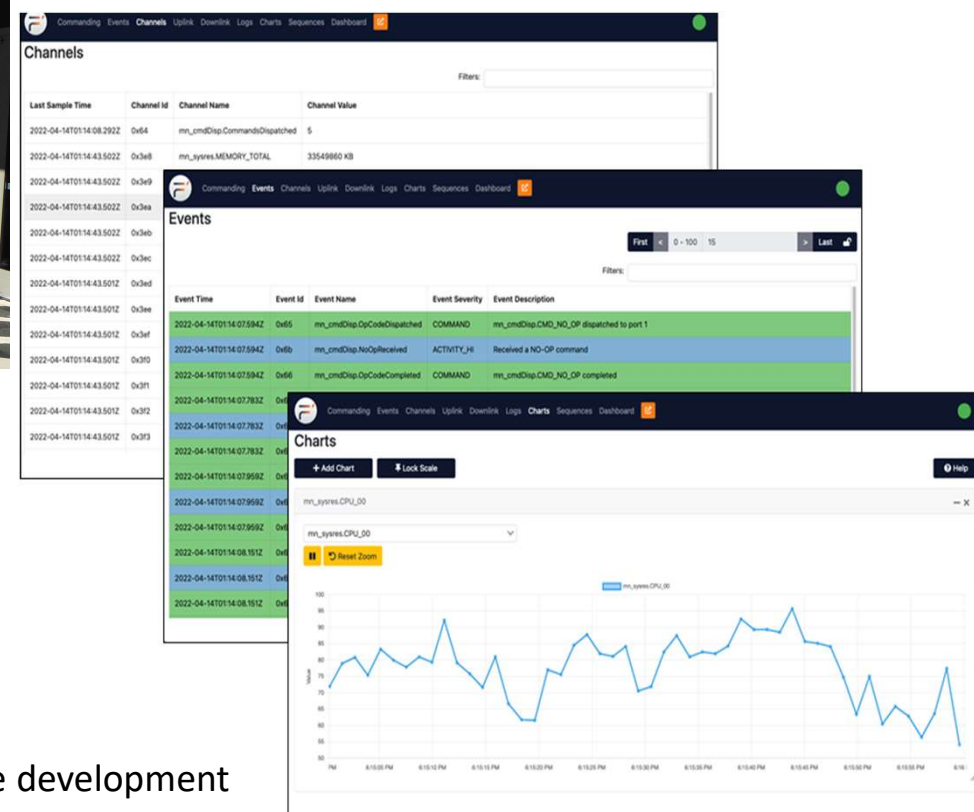
Cybersecurity

Roadmap: Maintenance, Tooling, and Support



Discussions

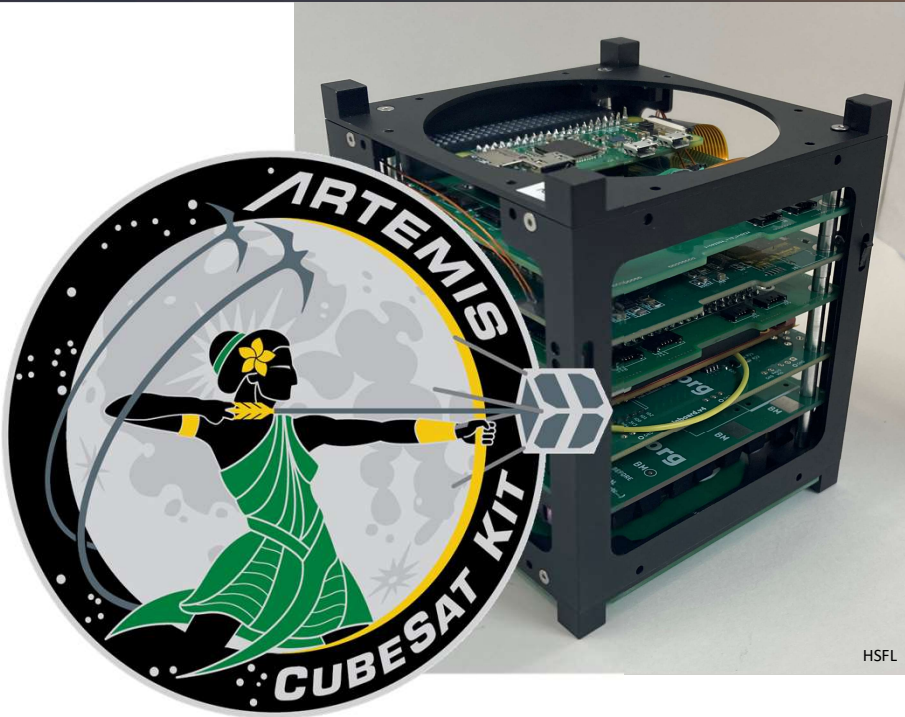
-   "Exiting with abort signal and core dump file" when using UART drivers and "startReadThread"
alimosallae19 asked 3 weeks ago in Q&A - **Answered**   22
-   Cap on the number of ports for "staticMemory.bufferAllocate"
alimosallae19 asked 2 weeks ago in Q&A - **Answered**   1
-   Analysis of stack size for Active components
sobkullr started 2 weeks ago in Ideas   6
-   How other missions handle finite state machines in F'?
sobkullr asked 2 weeks ago in Q&A - **Answered**   2
-   Clarification on array in FPP
SMoretini asked 2 weeks ago in Q&A - Unanswered   1



F' will continue to maintain the framework and tools that enable development and provide support to users.



Roadmap: Baremetal Best Practices



F' will provide guidance and recommendations to users deploying F' on systems without operating systems.

```
Deployment: Ref
-----
Size for F' Components
-----
.bss (Bytes)  Component
16            Ref::cycleLock
224           Ref::fatalHandler
224           Ref::linuxTime
224           Ref::textLogger
480           Ref::fatalAdapter
608           Ref::rateGroupDriverComp
1000          Ref::downlink
1144          Ref::fileUplinkBufferManager
1296          Ref::comm
1328          Ref::SG1
1328          Ref::SG2
1328          Ref::SG3
1328          Ref::SG4
1328          Ref::SG5
1336          Ref::blockDrv
1432          Ref::recvBuffComp
1448          Ref::fileManager
1448          Ref::pingRcvr
1544          Ref::fileUplink
1568          Ref::typeDemo
1616          Ref::systemResources
1928          Ref::sendBuffComp
2208          Ref::eventLogger
2480          Ref::rateGroup1Comp
2480          Ref::rateGroup2Comp
2480          Ref::rateGroup3Comp
2768          Ref::cmdSeq
3104          Ref::fileDownlink
3304          Ref::uplink
5464          Ref::prmdB
9336          Ref::staticMemory
11992         Ref::cmdDisp
14744         Ref::health
19776         Ref::tlmSend
-----
Minimum F' Configurations
-----
=== Number of Telemetry Channel Hash Slots (config/TlmChanImplCfg.hpp:45)
    - TLMCHAN_NUM_TLM_HASH_SLOTS = 21
=== Number of Telemetry Channel Buckets (config/TlmChanImplCfg.hpp:50)
    - TLMCHAN_HASH_BUCKETS = 86
=== Number of Commands (config/CommandDispatcherImplCfg.hpp:14)
    - CMD_DISPATCHER_DISPATCH_TABLE_SIZE = 80
-----
Size for Linux
-----
text  data  bss  dec  hex filename
1026253 23608 116600 1166461 11cc7d /home/echee/fprime-projects/fprime/Ref/build-artifacts/Linux/Ref/bin/Ref
```




Roadmap: Class-B Software Enhancements

Function / Data Usage			
17	Are declarations grouped into global and static/local?	Yes	Yes
18	Are all variables locally defined unless global or specific visibility is required?	Yes	Yes
19	Are global declarations uniquely named and namespaced?	Yes	Yes
20	Is the code free of any literals that are not properly documented in the form of macros or static constants (i.e. magic numbers)?	Yes	Yes
21	Have hardware specific code/data been sufficiently documented?	n/a	n/a
22	Is the precision of floating point numbers sufficient to ensure accuracy and floating-point equivalence is evaluated with margin (i.e. safe floating-point usage), and is not used in a loop control?	n/a	n/a
23	Are all variables initialized before use?	Yes	Yes
24	Have all non-atomic data elements been protected from corruption?	Yes	Yes
25	Have all function input arguments been validated prior to use?	Yes	Yes
26	Have unused arguments to functions been documented?	Yes	Yes
27	Verify that all commands are acknowledged success/failure	n/a	n/a
28	Verify that every EVR includes protection from cyclic generation. For persistent error conditions, the pattern should be a throttled EVR, and the total count of errors should be reported in EHA.	n/a	n/a
29	Verify EVR arguments match format strings. E.g. no U&U16s passed to %d.	n/a	n/a
30	Verify all counter EHA channels pushed with initial values at module init or task preamble.	n/a	n/a



Some checks haven't completed yet [Hide all checks](#)

JPL Coding Standard Scan / Analyze (cpp, jpl-standard-pack-2.yml) (pull_reque...

[Details](#)

JPL Coding Standard Scan / Analyze (cpp, jpl-standard-pack-3.yml) (pull_reque...

[Details](#)

F' will formalize the built-in quality by meeting NASA Class-B flight software standards.

Svc::Deframer (Passive Component)

1. Introduction

`Svc::Deframer` is a passive component. It accepts as input a sequence of byte buffers, which typically come from a ground data system via a `byte stream driver`. It interprets the concatenated data of the buffers as a sequence of uplink frames. The uplink frames need not be aligned on the buffer boundaries, and each frame may span one or more buffers. `Deframer` extracts the frames from the sequence of buffers. For each complete frame *F* received, `Deframer` validates *F* and extracts a data packet from *F*. It sends the data packet to another component in the service layer, e.g., an instance of `Svc::CommandDispatcher`, `Svc::FileUplink`, or `Svc::GenericHub`.

Roadmap: Cybersecurity



Bill of Materials



Uplink Encryption



Randomized Opcodes



Cybersecurity Standard
Operating Procedure

F' will bring safe cybersecurity practices to the forefront of the product line's development.





Questions?



NASA/JPL-Caltech



Further Information on F Prime

F Prime Website:

<https://fprime.jpl.nasa.gov>

F Prime Software:

<https://github.com/nasa/fprime>

F Prime Community:

<https://github.com/nasa/fprime/discussions>



Jet Propulsion Laboratory
California Institute of Technology

jpl.nasa.gov