

# Lesson 2

## Objectives

- Introduce the payload manage app, payload simulator app, and payload simulator library.
- Use the lib/app as examples for using the cFS App Exchange

# Library and Application Summary

## **PL\_SIM\_LIB**

- Simulate payload power states, detector states, and detector science data
- Provides an interface to set and clear a detector fault. Science data is corrupted when the fault is present
- JSON initialization table defines number of 1Hz cycles for power initialization and detector reset

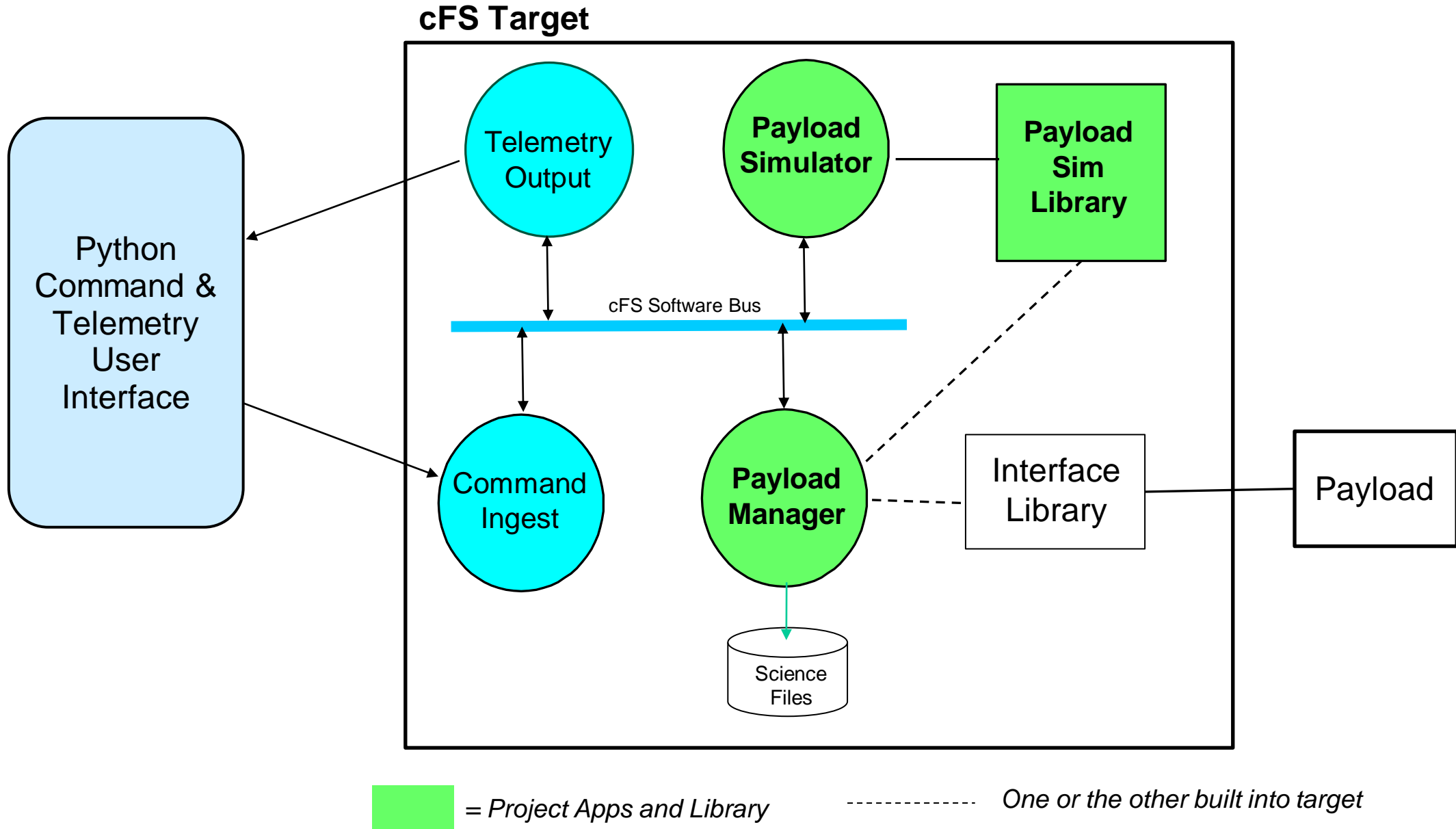
## **PL\_SIM App**

- Provides a ground command and telemetry interface to PL\_SIM\_LIB
- Command include: power on, power off, set fault, and clear fault

## **PL\_MGR**

- Manage the data interface to the payload and the creation of science data files
  - Reads detector data and writes images to files
- Commands to start and stop science data that turn on and off the detector, respectively
- JSON initialization table defines the science file path, base science filename and number of images per file

# Library and Application Architecture



# Install Payload Manager

- Use the steps described in lesson one to download, install and build PL\_MGR, PL\_SIM and PL\_SIM\_LIB
  - PL\_SIM\_LIB must be added to the cFS target first which causes it to be loaded before the apps when the cFS initializes
  - The PL\_MGR and PL\_SIM apps require PL\_SIM\_LIB global symbols to be defined when they are loaded

Select one or more apps to download then follow the steps in 'Add App'. See 'Add App' tutorial if you are unfamiliar with the steps.

- ☐ BC42\_CTRL Executes a 42 Simulator spacecraft control algorithm that has been ported from 42. This app requires the BC42\_LIB library, the E
- ☐ BC42\_INTF Provide an interface between a 42 Simulator socket and the cFS Software Bus.
- ☐ BC42\_LIB Encapsulates 42 simulator data types and functions as a single cFS library. It contains a minimal set of utilities that are required t
- ☐ BERRY\_IMU Core Flight System app that interfaces to the Ozzmaker Berry IMU over a Raspberry Pi I2C interface, <https://ozzmaker.com/berryi>
- ☐ GPIO\_DEMO Example app controlling a GPIO pin that can be used as a starting point for more sophisticated apps.
- ☐ MQTT\_GW Provide a gateway between the core Flight System (cFS) Software Bus messages and MQTT messages. The initial version sup
- ☐ MQTT\_LIB The Eclipse Paho Embedded C MQTT Library ported to the core Flight System. The Eclipse Paho software and the accompany
- ☐ PI\_IOLIB cFS library providing an interface to Raspberry Pi peripherals
- ☐ PL\_MGR Example payload management app.
- ☐ PL\_SIM Payload simulator app that provides a ground interface to the payload simulator library (PL\_SIM).
- ☐ PL\_SIM\_LIB Payload simulator library that simulates a fictitious detector that provides text data. PL\_SIM\_LIB relies on the payload simulator a
- ☐ SC\_SIM Simulate a simple spacecraft operational interface that exposes users to a remote operational interface. This is not a dynamic sir
- ☐ TBL\_SAT Raspberry Pi app used in the Table Sat kit.




Download

Cancel

# Running Payload Manager (1 of 2)

- After you start the new cFS target scroll through the start messages and verify that PL\_SIM\_LIB, PL\_SIM and PL\_MGR were loaded and initialized without any errors

cFS Target Process WindowTelecommand: 127.0.0.1:1234Telemetry: LocalTime: 1001051

Payload Simulator Library Initialized. Version 1.0.0  
1980-012-14:03:20.37583 CFE\_ES\_ParseFileEntry: Loading file: /cf/pl\_sim.so, APP: PL\_SIM  
1980-012-14:03:20.37597 CFE\_ES\_ParseFileEntry: Loading file: /cf/pl\_mgr.so, APP: PL\_MGR  
1980-012-14:03:20.42601 CFE\_EVS\_Register: Filter limit truncated to 8  
EVS Port1 66/1/KIT\_SCH 4: JSON initialization file successfully processed with 14 parameters  
1980-012-14:03:20.42620 CI\_LAB listening on UDP port: 1234  
EVS Port1 66/1/CI\_LAB\_APP 3: CI Lab Initialized. CI Lab App DEVELOPMENT BUILD v2.4.0-rc1+dev46,  
EVS Port1 66/1/FILE\_XFER 4: JSON initialization file successfully processed with 11 parameters  
EVS Port1 66/1/FILE\_XFER 100: FILE\_XFER App Initialized. Version 1.0.0  
EVS Port1 66/1/KIT\_TO 4: JSON initialization file successfully processed with 25 parameters  
EVS Port1 66/1/PL\_MGR 4: JSON initialization file successfully processed with 11 parameters  
EVS Port1 66/1/PL\_MGR 100: PL\_MGR App Initialized. Version 1.0.0  
EVS Port1 66/1/PL\_SIM 4: JSON initialization file successfully processed with 7 parameters  
EVS Port1 66/1/PL\_SIM 100: PL\_SIM App Initialized. Version 1.0.0  
EVS Port1 66/1/APP\_C\_DEMO 4: JSON initialization file successfully processed with 17 parameters

# Running Payload Manager (2 of 2)

These telemetry screens represent the default states

## PL\_MGR App Status Telemetry

PL_SIM/Application/STATUS_TLM - Port 9003		
App ID: 110	Length: 18	Seq Cnt: 63
Payload		
StatusTlm.Payload.ValidCmdCnt	:	0
StatusTlm.Payload.InvalidCmdCnt	:	0
StatusTlm.Payload.LibPowerState	:	OFF
StatusTlm.Payload.LibPowerInitCycleCnt	:	0
StatusTlm.Payload.LibDetectorResetCycleCnt	:	0
StatusTlm.Payload.LibDetectorState	:	OFF
StatusTlm.Payload.LibDetectorFault	:	FALSE
StatusTlm.Payload.LibDetectorReadoutRow	:	0
StatusTlm.Payload.LibDetectorImageCnt	:	0

## PL\_MGR App Status Telemetry

PL_MGR/Application/STATUS_TLM - Port 9004		
App ID: 109	Length: 81	Seq Cnt: 59
Payload		
StatusTlm.Payload.ValidCmdCnt	:	0
StatusTlm.Payload.InvalidCmdCnt	:	0
StatusTlm.Payload.PayloadPowerState	:	OFF
StatusTlm.Payload.PayloadDetectorFault	:	FALSE
StatusTlm.Payload.PayloadDetectorReadoutRow	:	0
StatusTlm.Payload.PayloadDetectorImageCnt	:	0
StatusTlm.Payload.SciFileOpen	:	FALSE
StatusTlm.Payload.SciFileImageCnt	:	0
StatusTlm.Payload.SciFilename	:	Undefined

For more information on payload manager refer to the following Basecamp project:

[https://openmissionstack.com/projects\\_read/payload\\_manager](https://openmissionstack.com/projects_read/payload_manager)