National Aeronautics and Space Administration

Core Flight Executive (cFS) Training

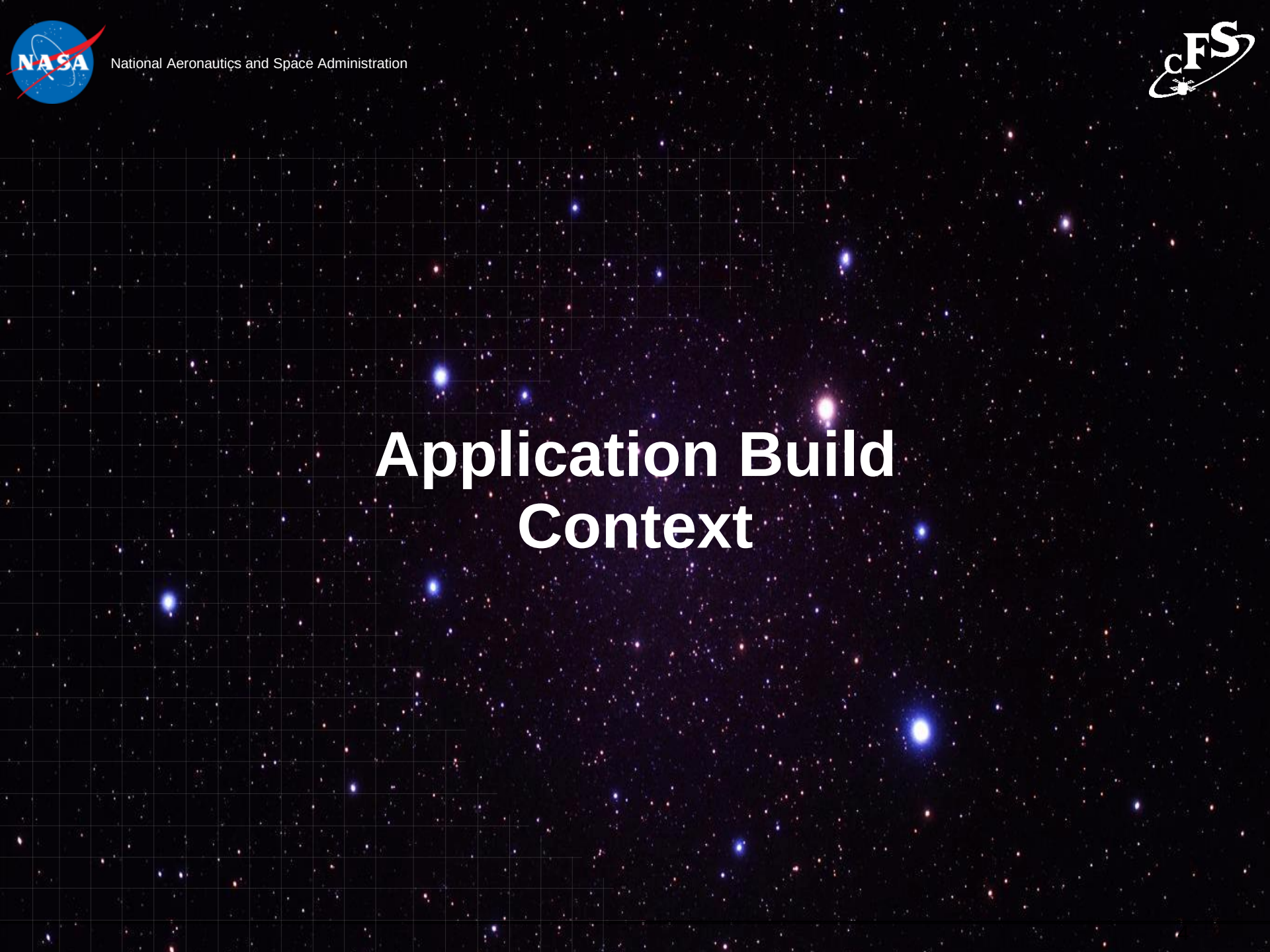
Application Development

Objectives and Intended Audience

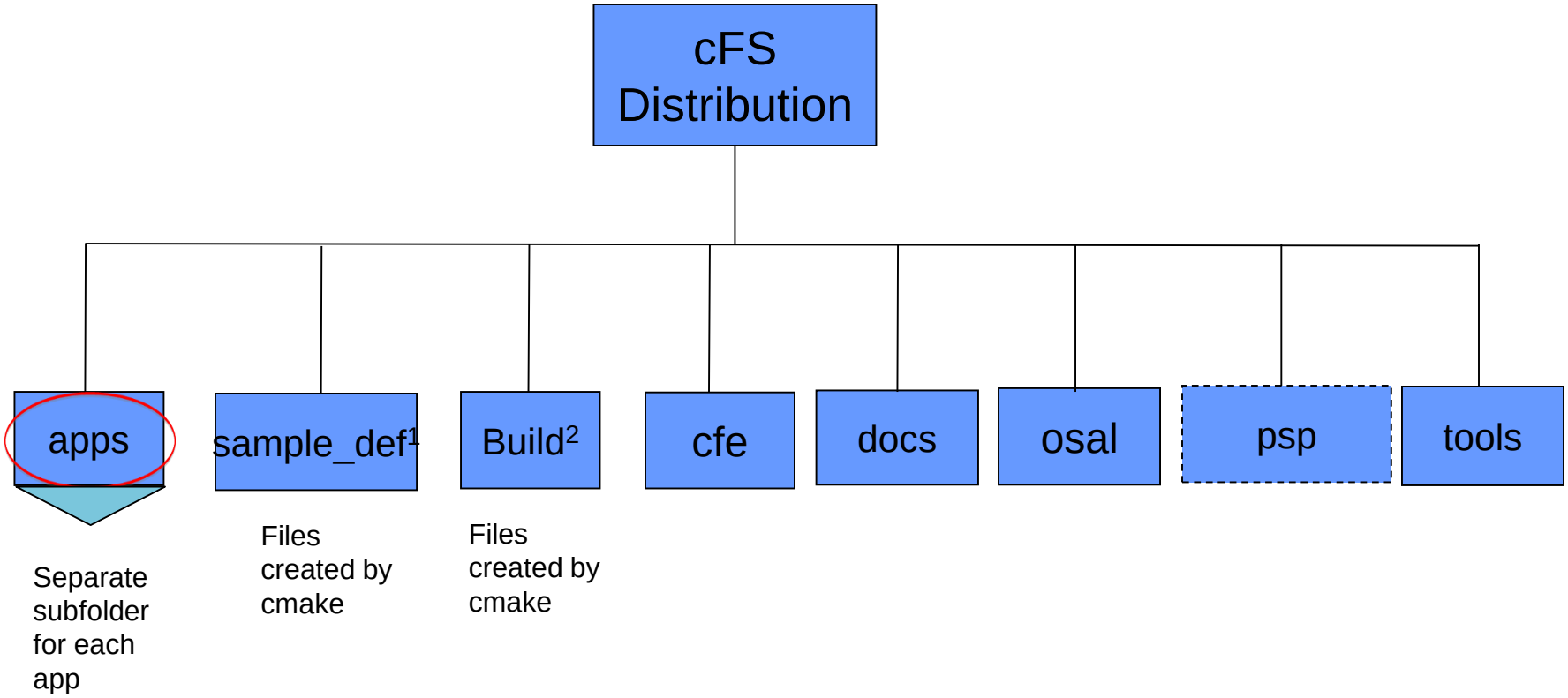
- **Describe application development/runtime environments and basic application design**
  - Spacecraft systems engineers and FSW engineers
- **Learn how to develop a new app using hands-on exercises**
  - FSW Engineers
  - Working knowledge of C language required

Agenda

- Application build context
- Application runtime context
- Application design
- Class Exercises

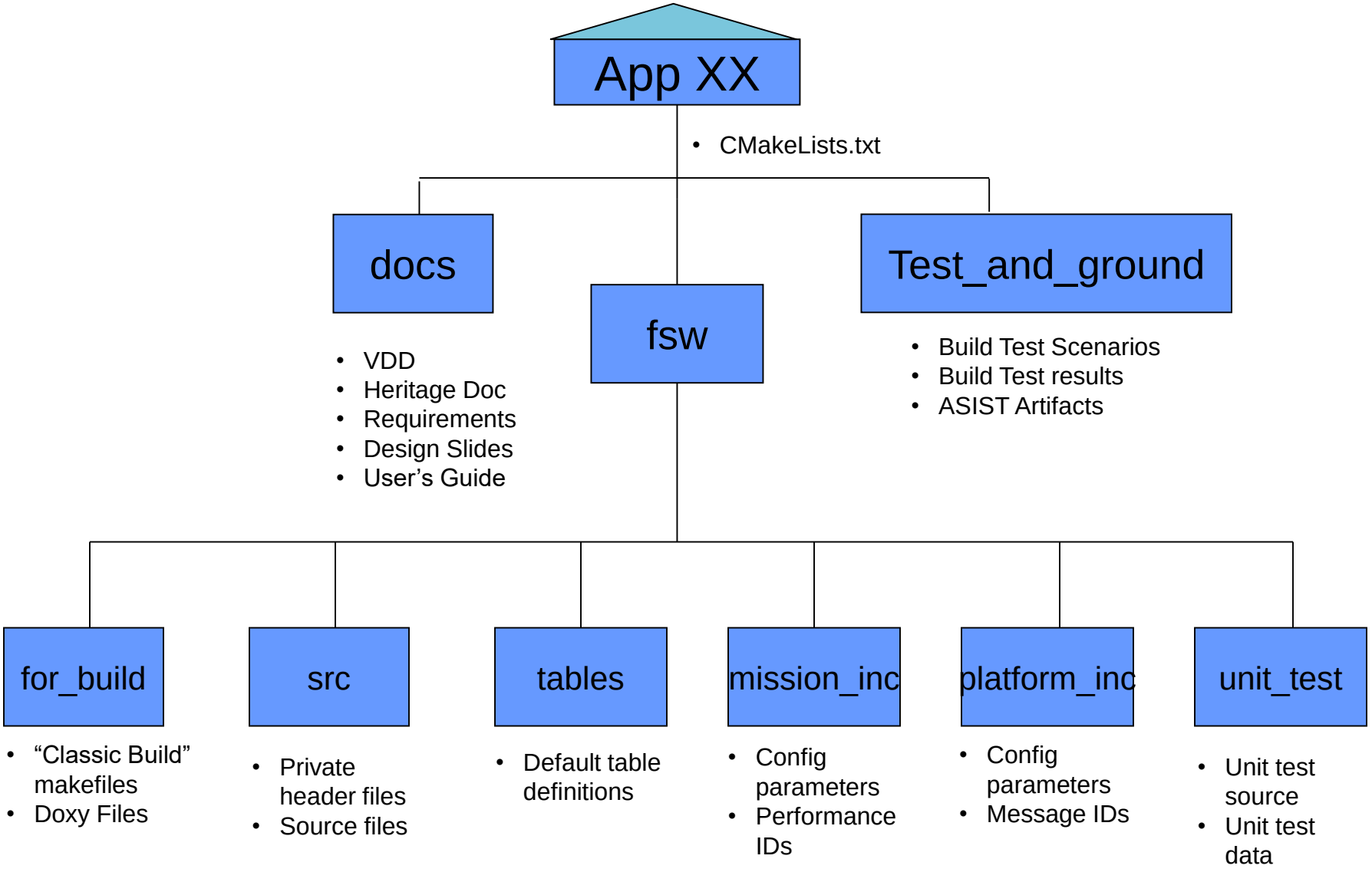

National Aeronautics and Space Administration
Application Build Context

cFS Mission Directory Structure



1. Initially copied from …/cfe/cmake/sample_def. Missions typically rename this directory
2. Files created by cmake

cFS Application Directory Structure



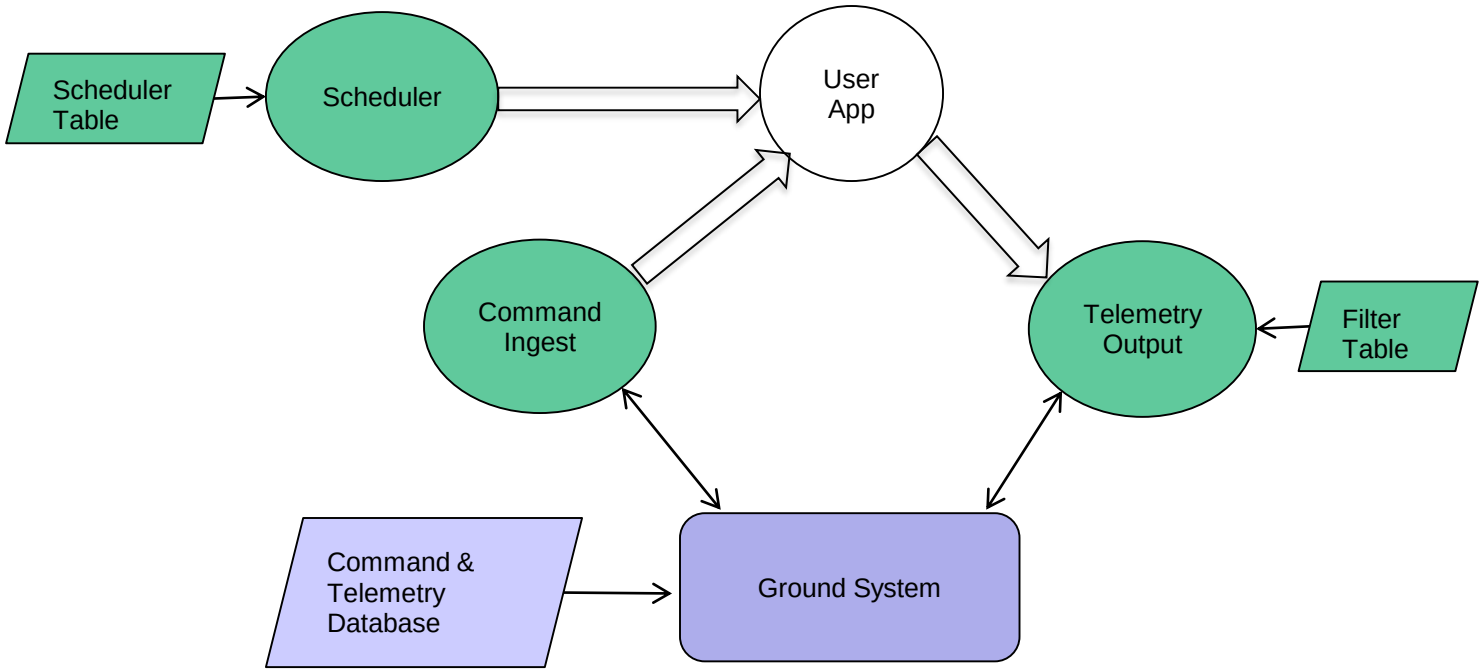
sample_def Directory

- **Targets.cmake**
  - Identifies the target architectures and configurations
  - Identifies the apps to be built
  - Identifies files that will be copied from sample_def to platform specific directories

- **Copied file examples**
  - cpu1_cfe_es_startup.scr
  - cpu1_msgids.h
  - Cpu1_osconfig.h


National Aeronautics and Space Administration
NASA
cFS
Application Runtime Context

Application Runtime Context



Application Runtime Context

- **Scheduler (SCH) App**
  - Sends software bus messages at pre-defined time intervals
  - Apps often use scheduled messages as wakeup signals
- **Command Ingest (CI) App**
  - Receives commands from an external source, typically the ground system, and sends them on the software bus
- **Telemetry Output (TO) App**
  - Receives telemetry packets from a the software bus and sends them to an external source, typically the ground system
- **SCH, CI, and TO provide a runtime context that can be tailored for a particular environment**
  - cFE delivered with 'lab' versions, CI & TO use UDP
  - JSC released CI and TO versions that use a configurable I/O library
  - OpenSatKit uses apps with text-based tables defined using JSON


National Aeronautics and Space Administration
NASA
cFS
Application Design

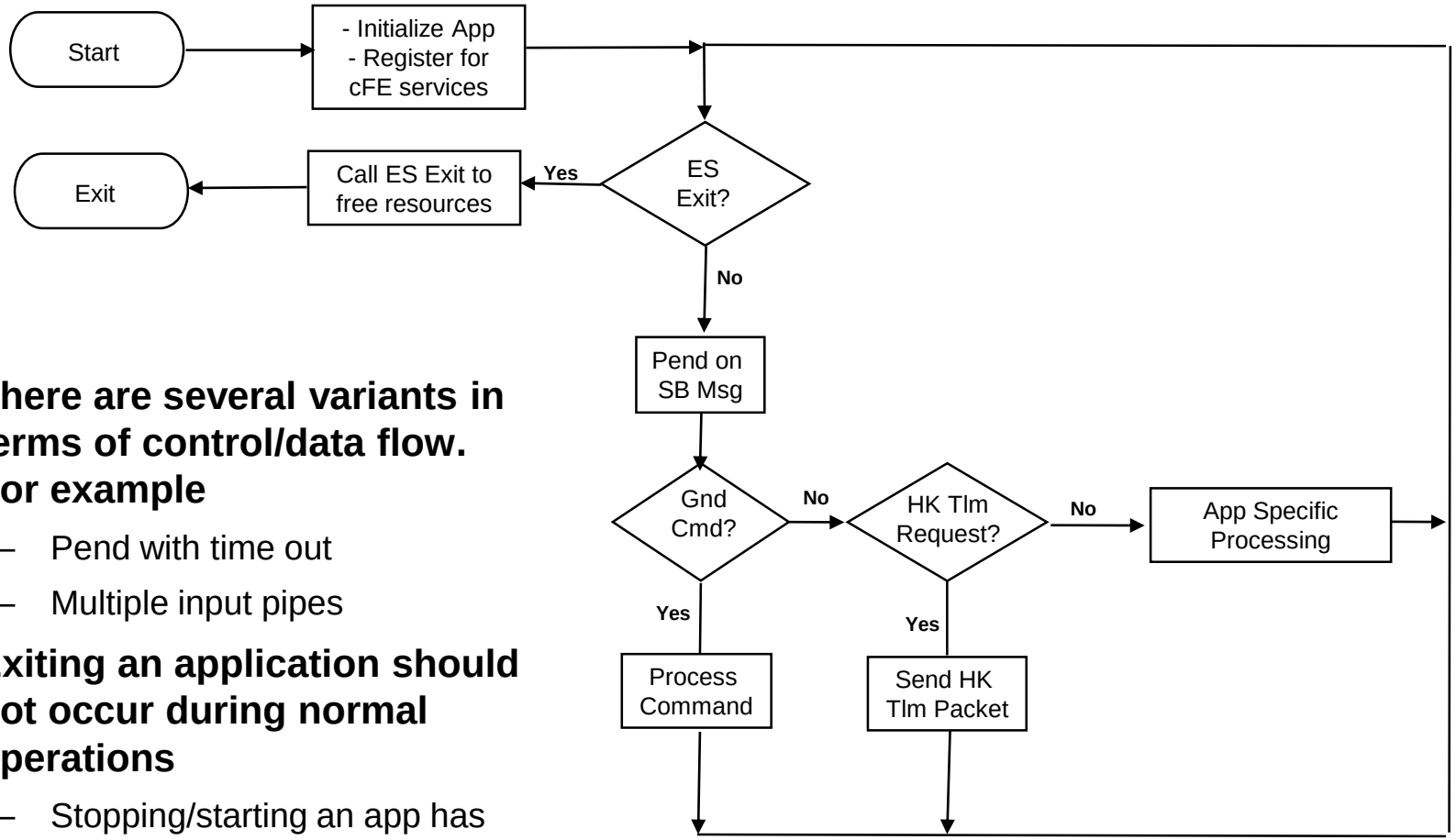
Application Design Intro

- **cFE/docs/cFE Application Developers Guide.doc**
  - Provides a good description of how to use cFE services/features
  - Provides one example of an application template

- **Application frameworks**
  - Organizations have created frameworks in C and C++ but they are not publically available
  - OpenSatKit contains an object-based app framework in C
    - Prototype that hasn't been used in flight

- **Application design patterns**
  - There are patterns but they have not been formally captured
  - When creating a new app look for an existing app that has similar operational context

Generic App Design

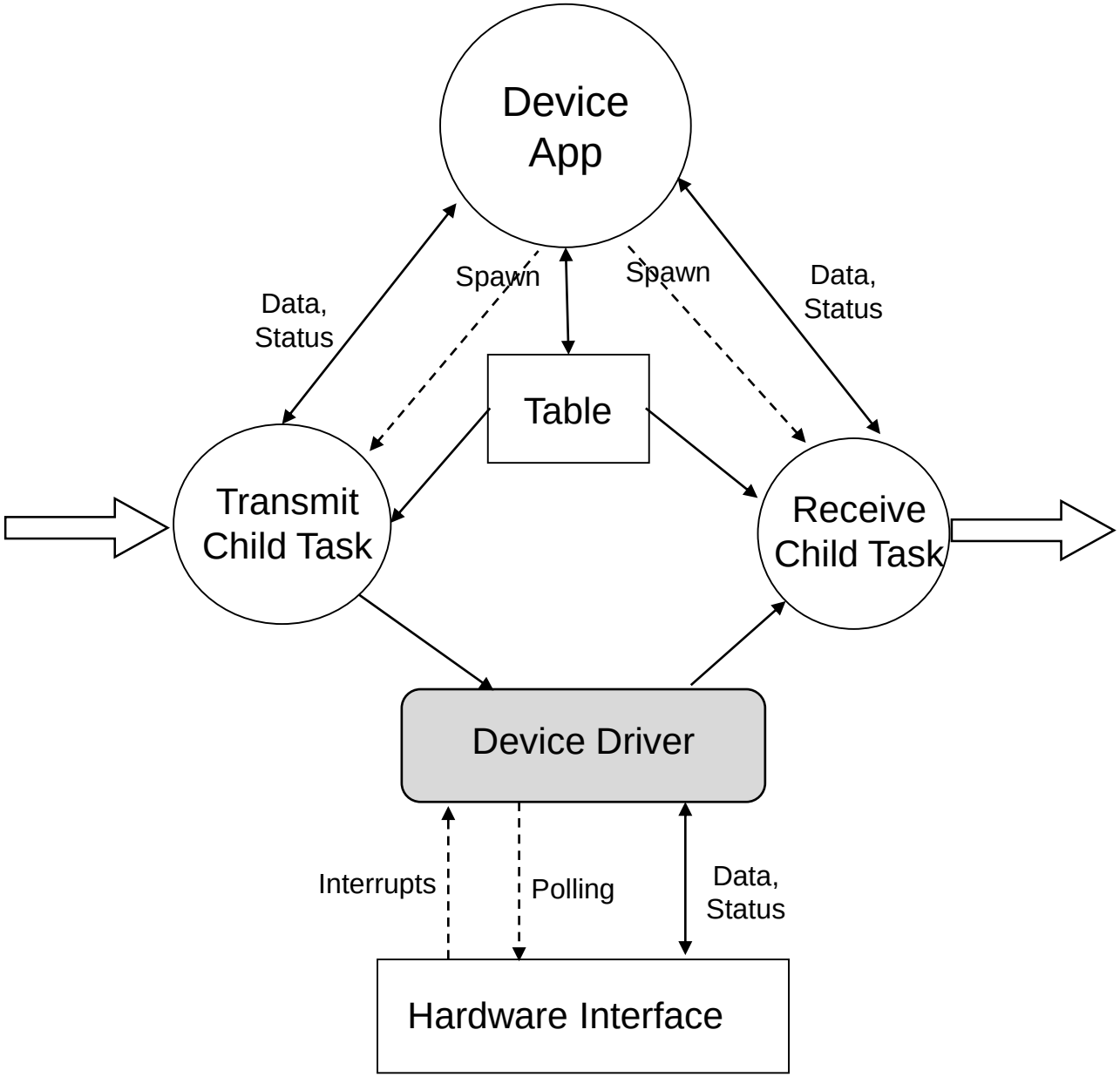


- **There are several variants in terms of control/data flow. For example**
  - Pend with time out
  - Multiple input pipes
- **Exiting an application should not occur during normal operations**
  - Stopping/starting an app has been used for in-orbit maintenance

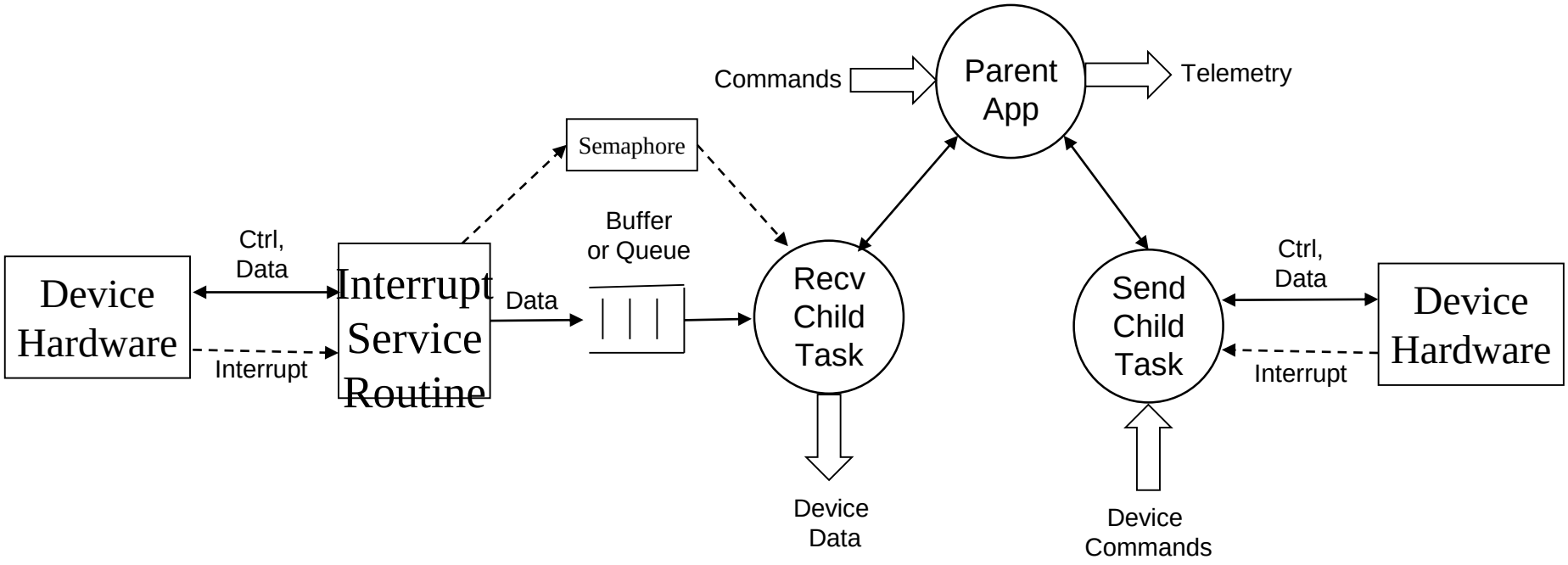
Application Design Practices

- **Allocate resources during initialization to help keep runloop deterministic**
- **Use a lower priority child task for long operations like a memory dump**
  - Create child tasks during initialization
- **Register with EVS immediately after registering app so local event log can be used instead of system log**
- **NOOP command sends an informational event message with app's version number**
- **Use SCH app to periodically send a "send housekeeping" message**
  - Housekeeping data includes command counters and general app status
  - 3 to 5 seconds is a common interval
  - Attitude Determination and Control apps don't typically use this pattern
- **"Hello World" app generation tools**
  - Multiple tools exist, but none have been sanctioned as demonstrating best practices

I/O Application Design Pattern



I/O Application Design Pattern



- **General control/data conceptual flow**
  - Each communication bus has a specific protocol
- **Architectural role**
  - Read device data and publish on software bus
  - Receive software bus messages and send to the device

National Aeronautics and Space Administration

App Development Exercises

See Supplemental Student Material