Success Oriented Ground and Space Software Defined Architectures

30 March 2004

Jerry Brand, Vince Kovarik and James Faist
Melbourne, Florida

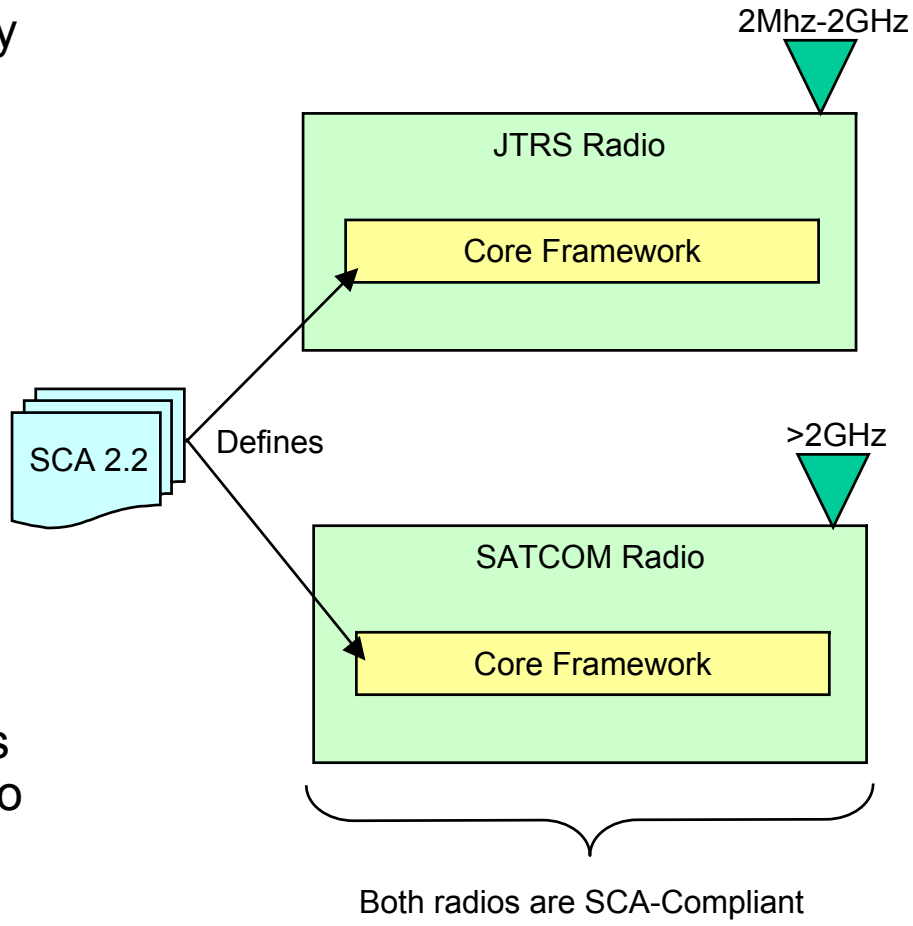
Presented at the GSAW 2004
Manhattan Beach, CA

Introduction

- Software Defined Radio (SDR) Architectures Background
- JTRS/SCA Terminology
- SCA Software Structure
- Example Ground SDR
- Example Flight SDR
- Successful Architecture Attributes
- Conclusions

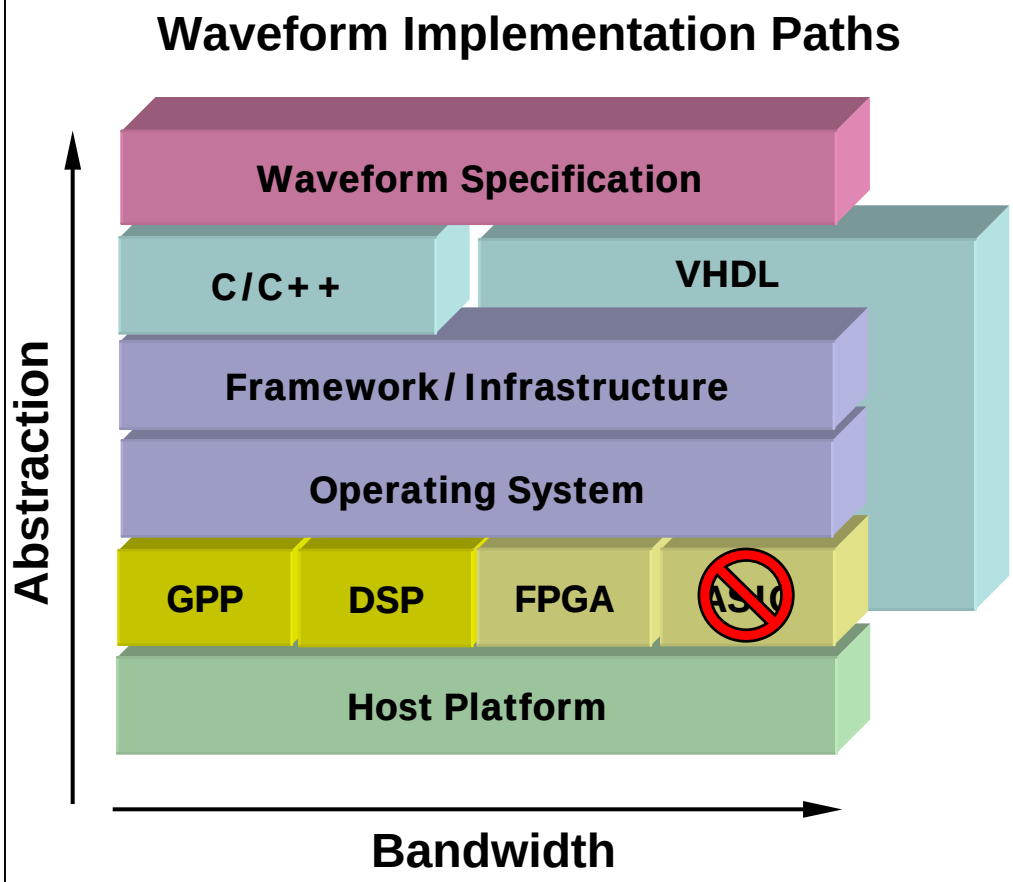
JTRS/SCA Terminology

- **Joint Tactical Radio System (JTRS):** A radio system within the 2MHz to 2Ghz (tactical) frequency range and has the SCA infrastructure (see below).
- **Software Communication Architecture (SCA):** The underlying infrastructure specification that must be implemented.
- **Core Framework (CF):** The realization of the SCA specification that provides the radio infrastructure services specified in the SCA.
- **SCA-Compliant:** A radio that has a Core Framework that adheres to the SCA is SCA-Compliant.

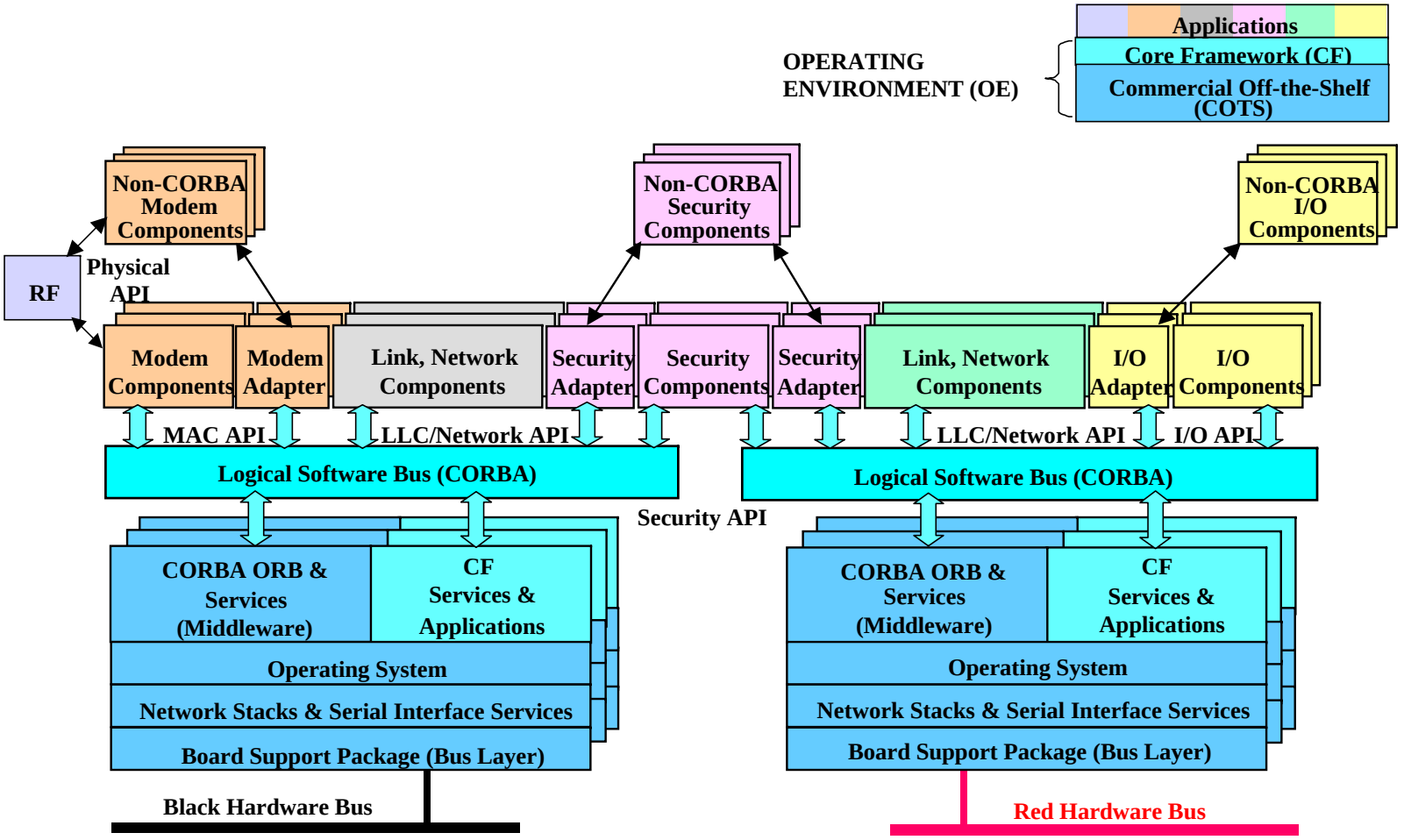


SDR Abstraction

- Dynamic (re)configurability
  - Components
  - Waveforms
  - Processing paths
- Plug-n-Play components
- Reuse of common implementations
- The Framework or Infrastructure provides the intelligent abstraction away from the physical implementation
- The Framework provides the foundational technology for realizing a cognitive radio.

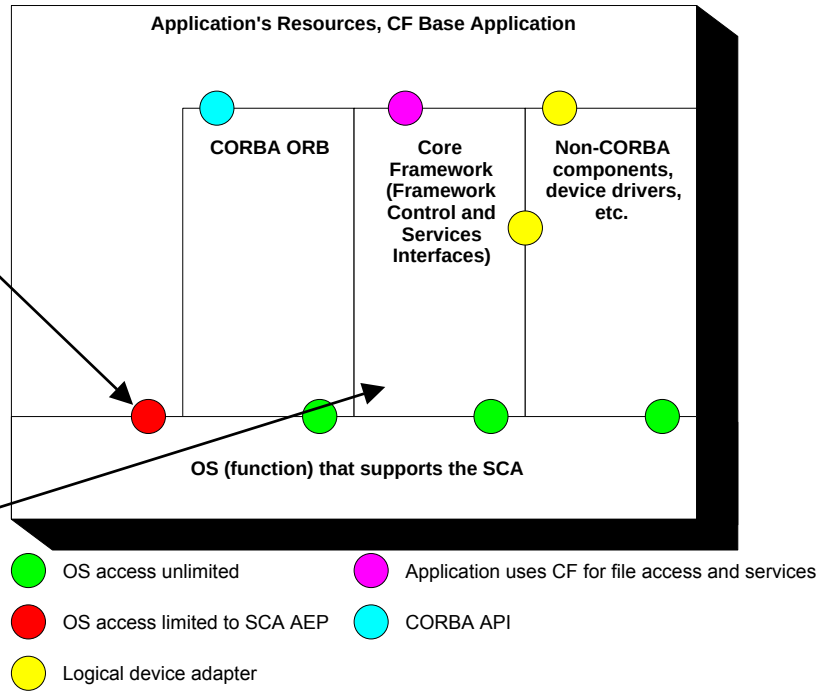


SCA Software Structure

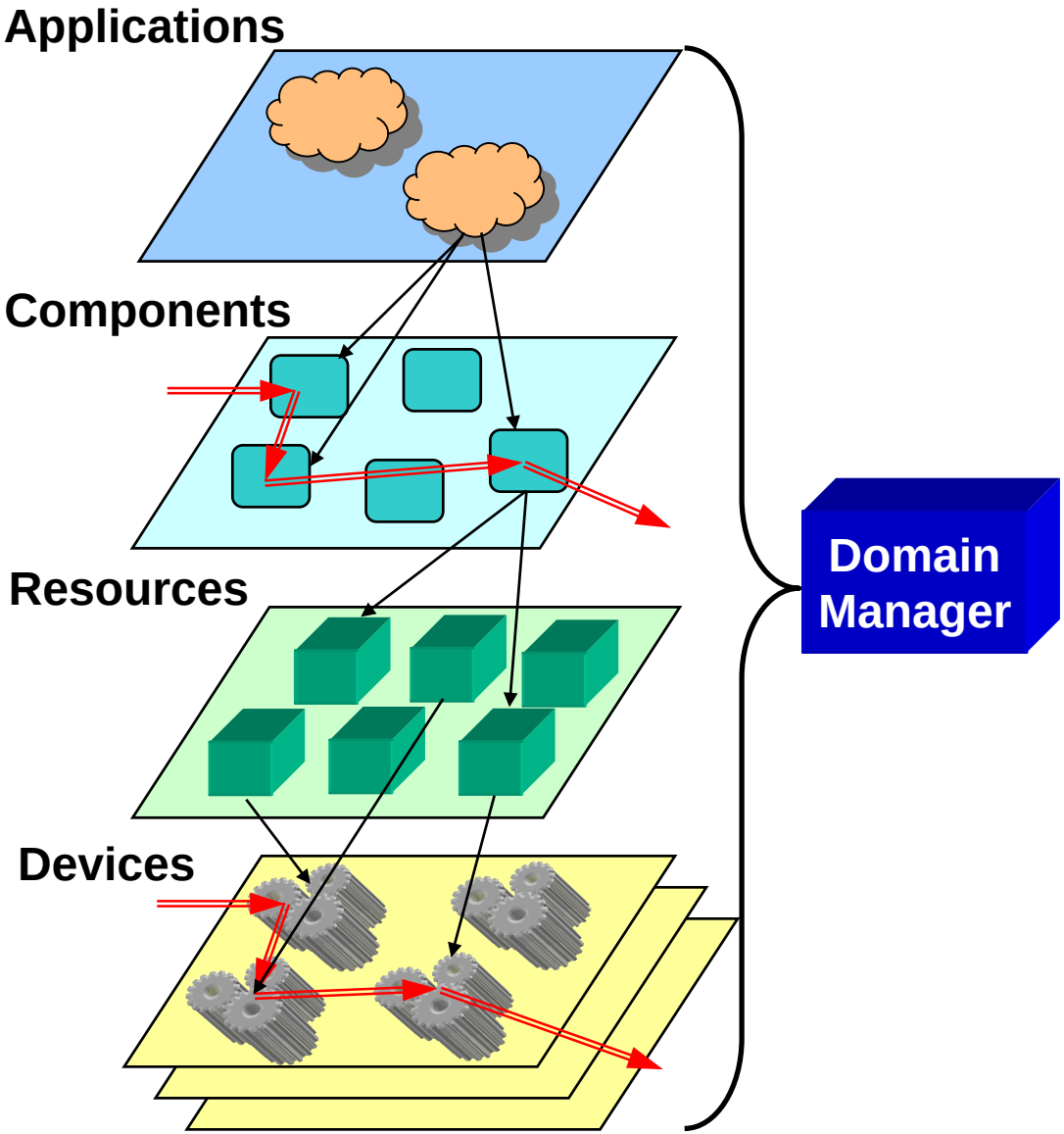


RTOS and Core Framework

- Real-Time Operating System (RTOS) Must Support the SCA Application Environment Profile (AEP)
  - The SCA AEP is a subset of the POSIX.13 Real-time Controller System Profile (PSE52)
  - Can be fully POSIX Profile 52 (or greater) compliant
- Applications shall be limited to using the RTOS services that are designated as mandatory in the SCA AEP
- The Core Framework implements a essential functionality of the SCA



SCA Concept Hierarchy



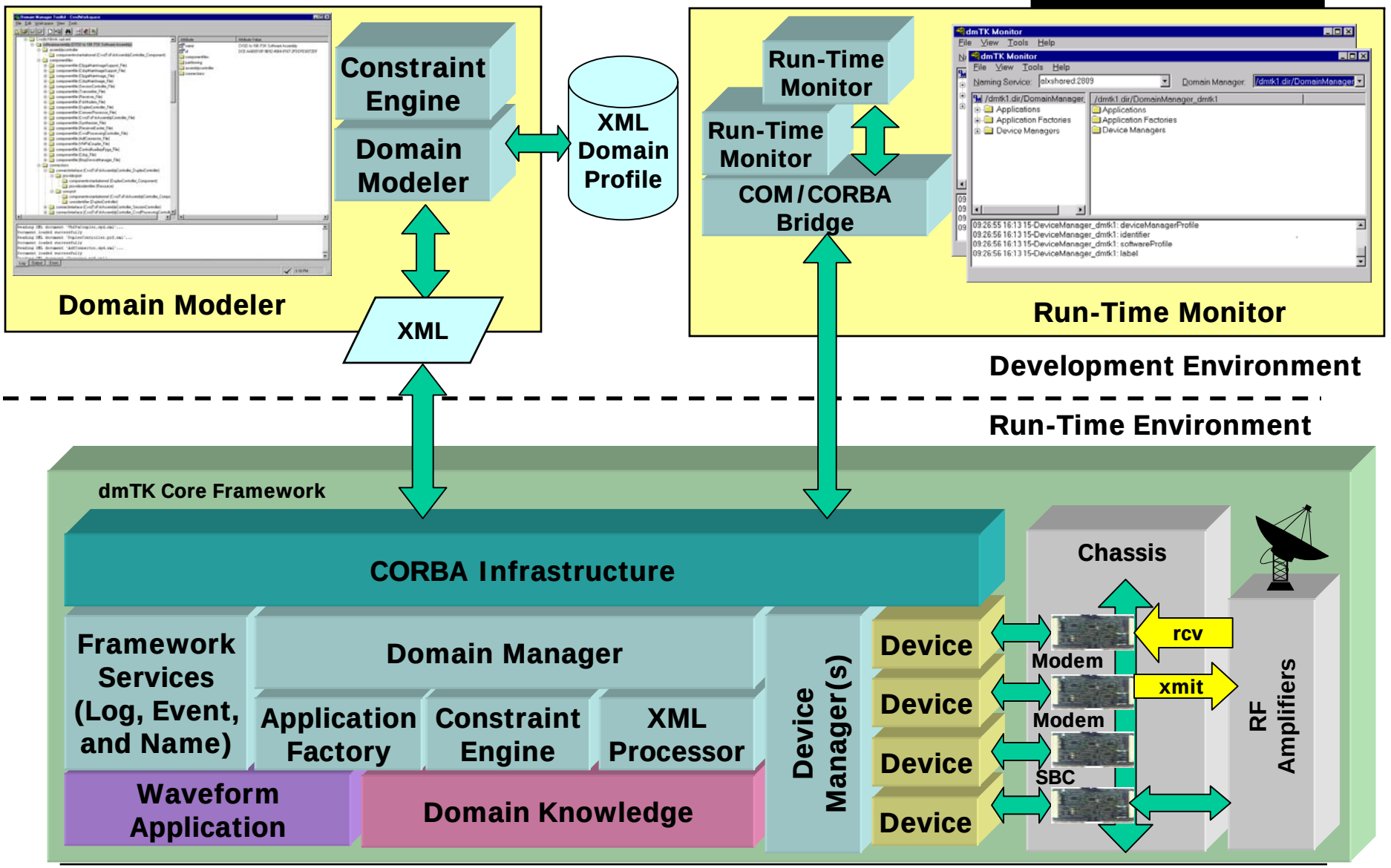
- **Application** – The top-level entity used by the radio operator, i.e. waveform.
- **Components** – Software units that provide specific functionality either directly or through underlying hardware.
- **Resources** – An abstraction of the type, capacity, and state of a logical entity.
- **Devices** – The collection of physical elements comprising the radio system.

Various Typical SCA Compliant Operating Systems and Platforms

HARRIS

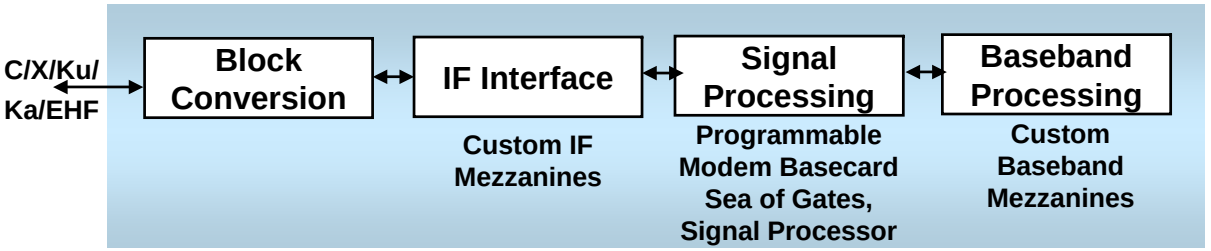
- SCA 1.1
  - VxWorks / Pentium / ORBexpress GT 2.1.4
  - VxWorks / PowerPC / ORBexpress GT 2.1.4
  - VxWorks / StrongARM / ORBexpress RT 2.3.1A-β
  - LynxOS / PowerPC / ORBexpress GT 2.1.4B-β
- SCA 2.2
  - Windows / Pentium / ORBexpress RT 2.3.5
  - Windows / Pentium / ACE-TAO
  - VxWorks / PowerPC / ACE-TAO
  - Linux / Intel / ACE-TAO
  - VxWorks / PowerPC / ACE-TAO (OCI)
  - Windows / Intel / ACE-TAO (OCI)

Domain Management



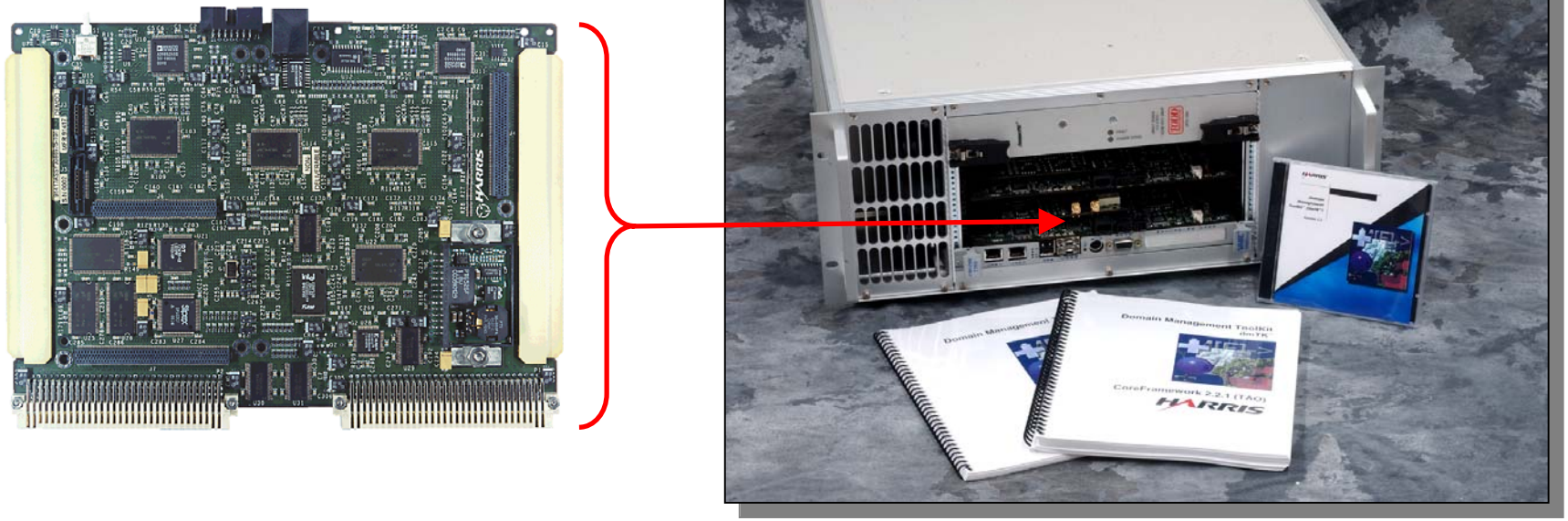
Software Definable Features for Space and Ground Architectures

- **Modulation Modes** – FSK, MSK, QPSK, OQPSK, SQPSK, SOQPSK, DPSK, SFDPSK, GMSK
- **IF Interfaces** – 70 MHz, 140 MHz, 700 MHz, L-Band 950 to 2050 MHz
- **Data Rates** - 9600 bits/second to 1 Gbit/sec basecard
  - Multiple Systems Supported
- **Forward Error Correction Coding**
  - Turbo (Rate 2/3, 3/4)
  - K=7 Convolutional (Rate 1/2)
  - Reed Solomon
  - Turbo (other rates)
- **Other Specialized Functions**
  - Bit Count Integrity
  - Mux/Demux for In-Band Control
  - Data Scrambling
  - Differential Encoding
  - Interleaving/ De-interleaving
  - Phase ambiguity resolution
- **Interfaces**
  - Control and Status
  - Local and Remote Operator
  - Ethernet, RS-232, RS422, RS485



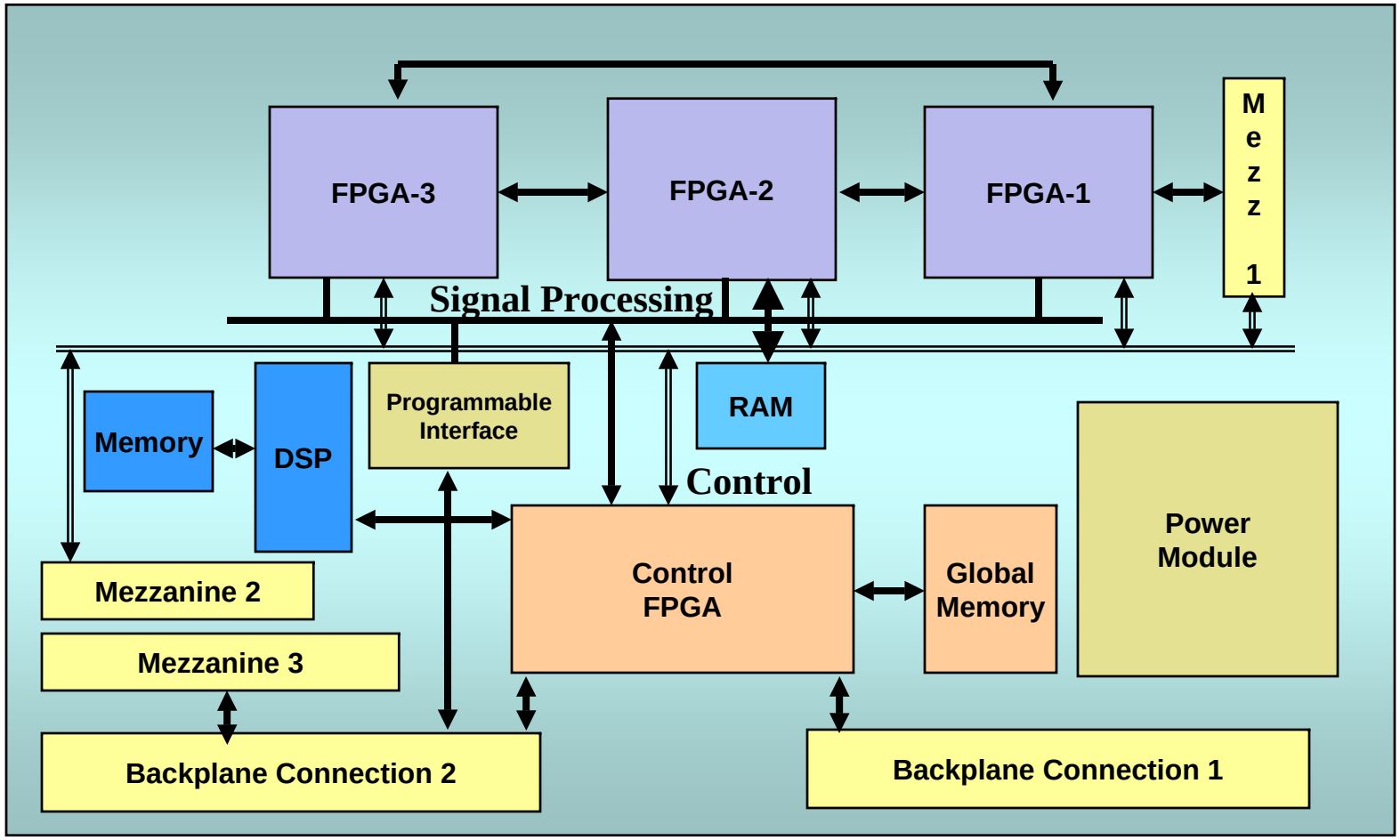
Algorithms are loaded and executed in programmable devices
Processor basecard supports a modulator/ demodulator application
Transmit or receive IF modules at different frequencies mount on the CCA
Customized mezzanine I/O modules provides maximum flexibility

SCA Compliant Programmable Modem Meets SDR Needs

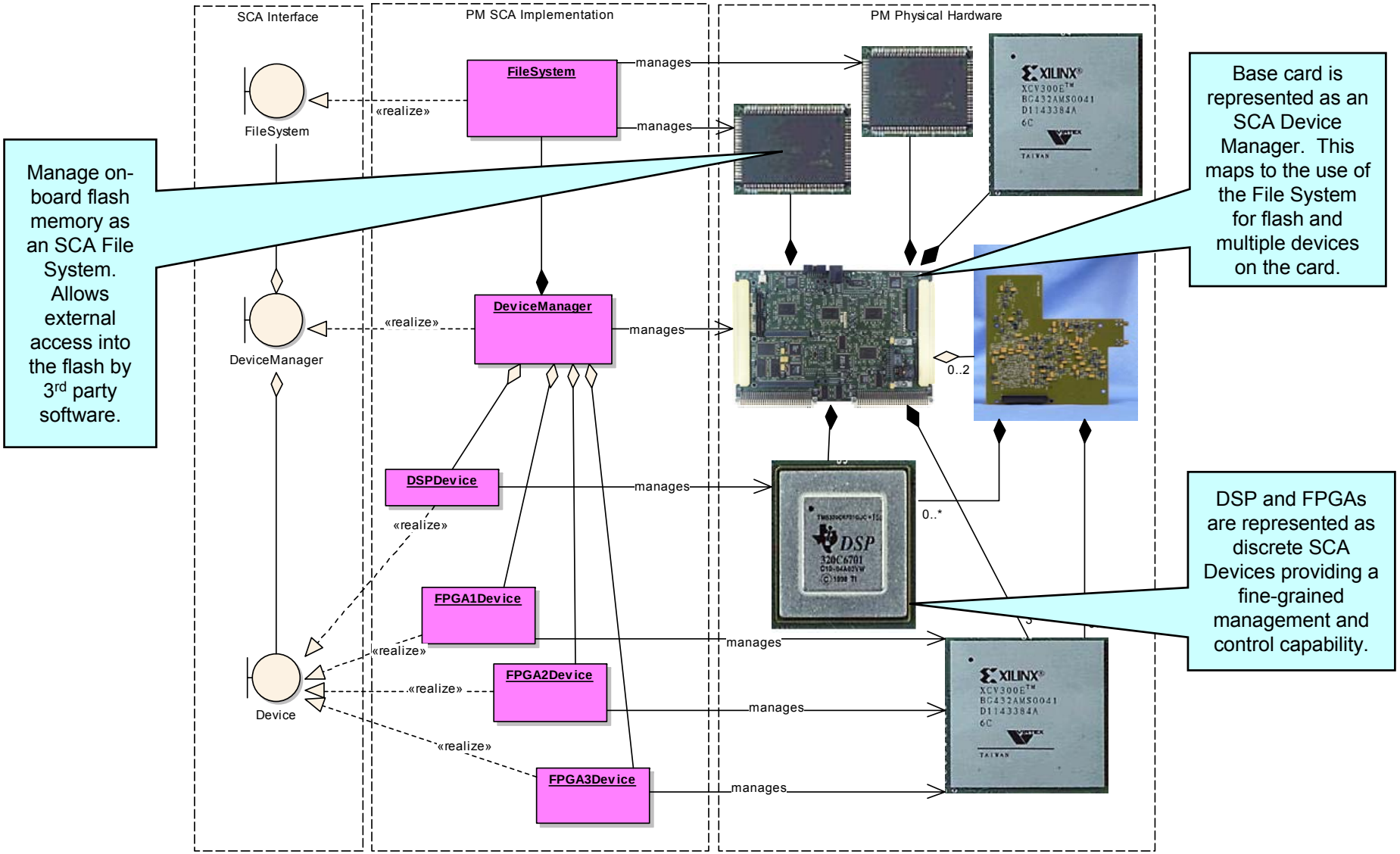


- **SCA Device and Device Manager interfaces capabilities include:**
  - Device Manager interface for the board
  - SCA File System interface for on-board flash
  - Logical Device interface for each FPGA and the DSP
- **Demonstrated SCA and SDR Programmability Capabilities**
  - Legacy compatibility
  - Control of external components (video for example)
  - Shutdown of TCDL waveform, load, and start of CDL waveform accomplished in under 10 seconds

Typical Ground SDR Base Card Structure



Modem SCA Architecture

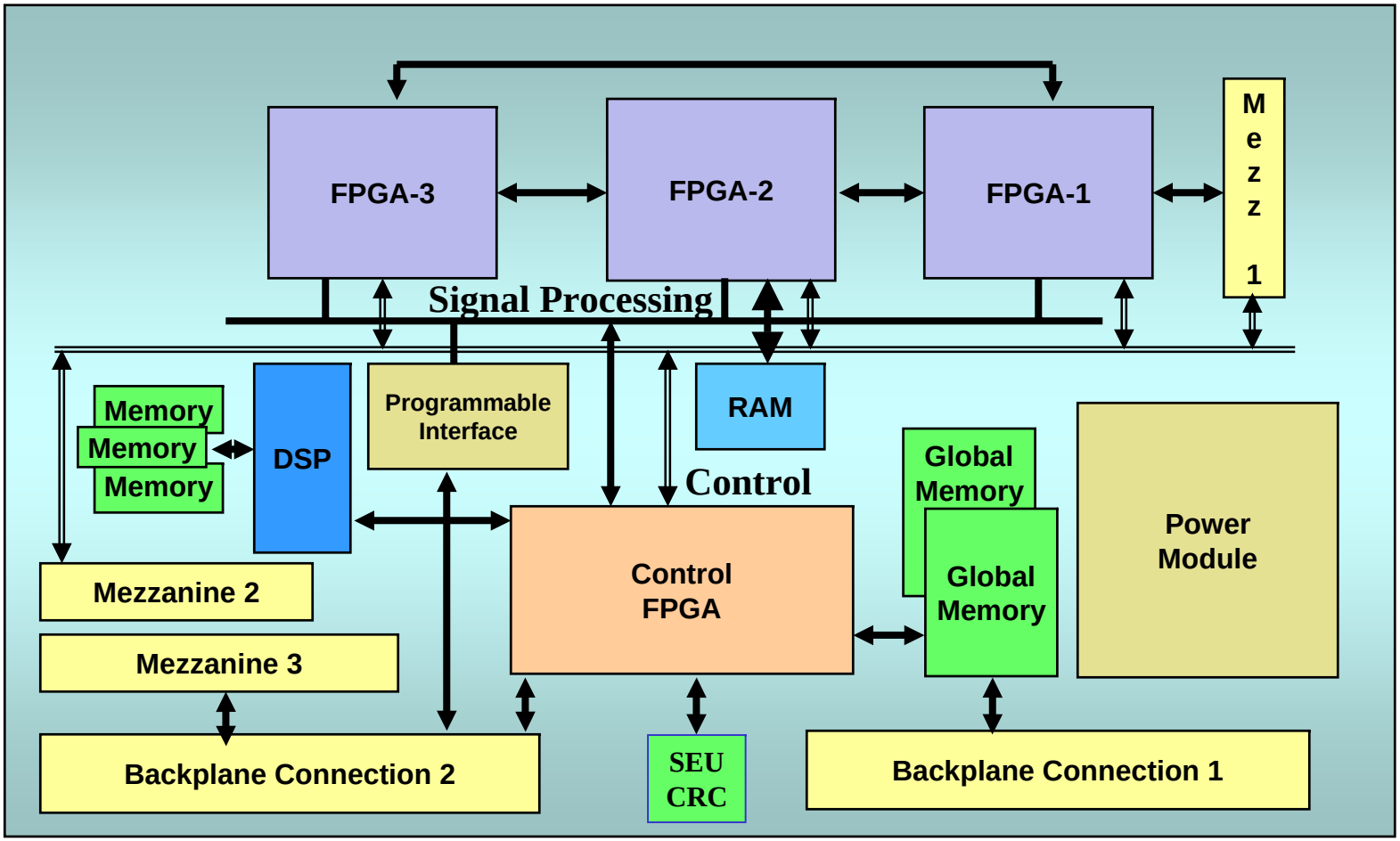


SDR Flight Processor Requirements Derived from the Ground Basis

- Multiple FPGAs
  - Maintain identical pin-out on all devices
    - Simple configuration creation/ partitioning
  - High Speed Communication busses
  - Single Microprocessor Interface
  - Independent banks of memory per FPGA
  - Built in Test for fault detection, some isolation, of assembly
    - Configuration
    - Opens
    - Shorts
    - Signal integrity
  - Implement a SEU detection scheme for configuration bit streams

These Requirements Support Successful in-Flight Software Programmability

Typical Space SDR Base Card Structure



Flight Architecture Builds Directly from the Ground Architecture

Successful Ground and Space Architecture Attributes

- Programmability is built-in from the start
- SCA compliance stems from a cohesive development and run-time environment design methodology
- SDR programmability must be accomplished seamlessly and in minimum time
- Space SDR platforms build directly from sound ground SDR architectures and features
- Joint optimization of ground and space architectures lead to a successful, truly programmable Ground/Space system

Conclusions

- Software defined implementations provide more opportunity across multiple platforms and maximize design reuse
- Reuse applies not only at multiple levels within an implementation, but across domains such as from ground to space
- Common open architectures deliver designs amenable to improvements by multiple parties
- Overall, SDR implementations provide substantial benefits and lead to successful ground and space communications architecture