

Configuration Management (CM)

A vital piece of Systems Engineering

A CM process shall exist to guide system products, processes, and related documentation, and to facilitate the development of open systems. The effort includes identifying, documenting, and auditing the functional and physical characteristics of an item, recording the configuration, and controlling changes to an item and its documentation. It shall provide a complete audit trail of decisions and design modifications.

DoD 5000.2R, C5.2.3.4.5



Configuration Management is a Cornerstone to the Systems Engineering Mission

- **CM “controls integrity” of the SE process:**
 - 1) *system components are identified and documented*
 - 2) *controls component change process*
 - 3) *maintains change processing and product data*
 - 4) *verifies compliance with specifications, etc.*

•References

- MIL-HDBK-61A Configuration Management Guidance (2/7/01)
- ISO 10007 Quality Management Guidelines for CM
- ANSI-STD-649 National Consensus Standard for CM
- MIL-STD-973 Configuration Management (cancelled for use on new programs)
- Configuration Management Critical Process Assessment Tool



Why Configuration Management?

- **Benefits**

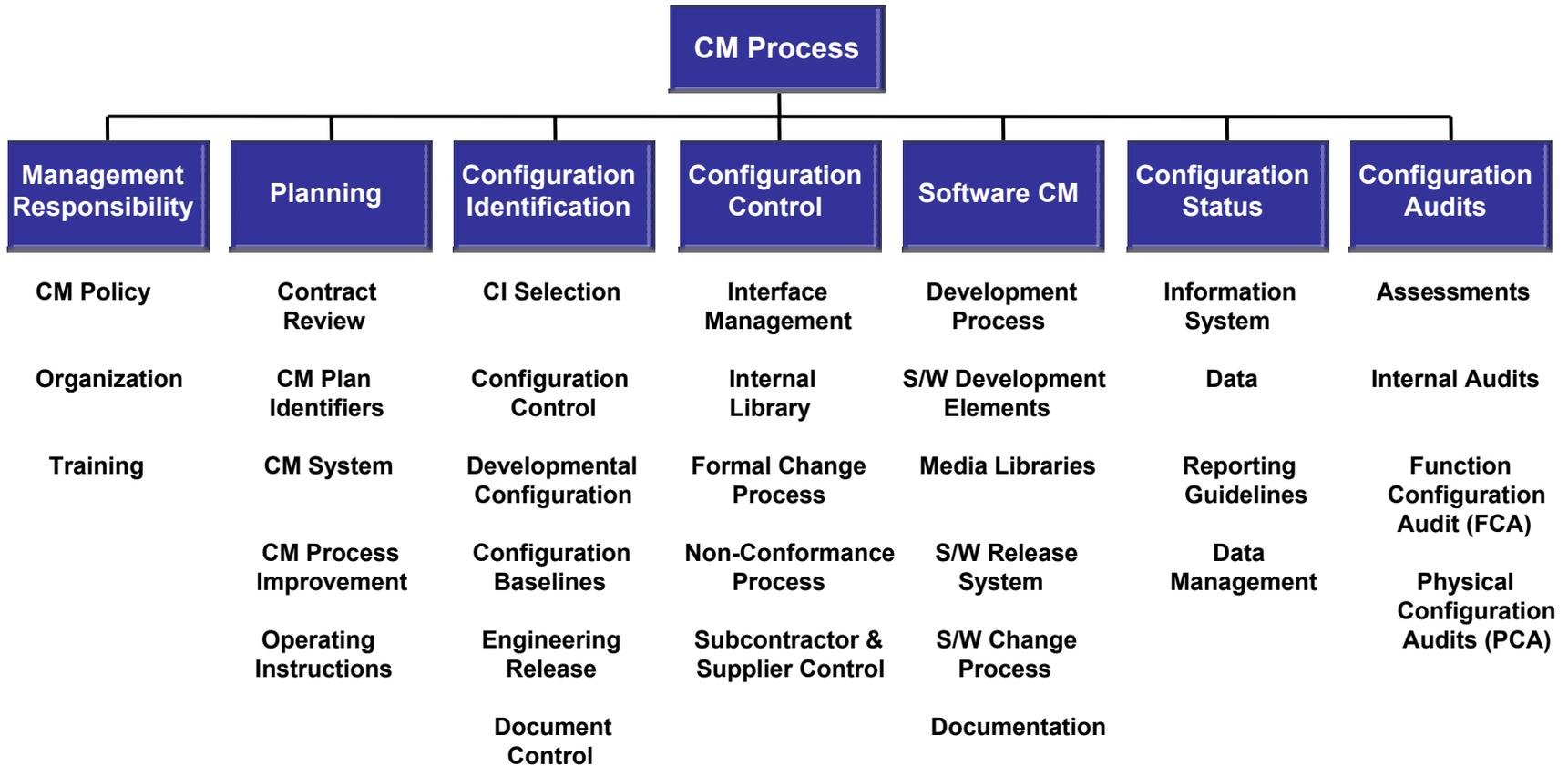
- System Attributes are fully defined and documented
- Change activity is managed
- Components / interfaces are appropriately identified
- Accurate status information reduces delays/down-time
- System is verified against the requirements prior to acceptance

- **Weak CM leads to:**

- Program delays & equipment failures
- Problems & increased costs due to hardware/software/interface inconsistencies
- Decreased System operational effectiveness & supportability
- The severest consequence which would be catastrophic loss of expensive equipment and human life.



CM Process Elements



Must be familiar with the implication of all listed elements (SMC/Internal & Contractor/External) to adequately assess the risk associated with respective CM decisions.



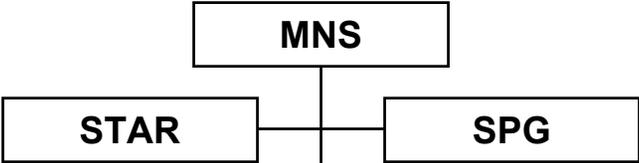
“Baseline” CM Concept

- **Established at key points in the Program life cycle**
 - System Baseline: *System specification and interface documents (usually established between SDR & PDR)*
 - Allocated Baseline: *Allocation of system level performance requirements and constraints to the element level (end of CDR)*
 - Product Baseline: *Allocation of element level performance requirements and constraints (end of EMD, after FCA & PCA)*

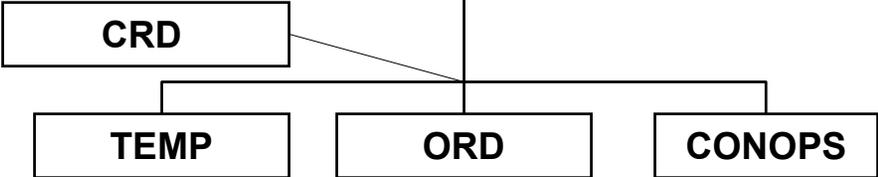


DoD Document Tree

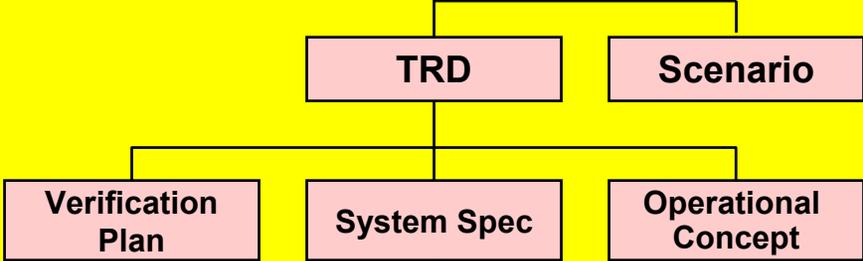
Mission
Needs
Related



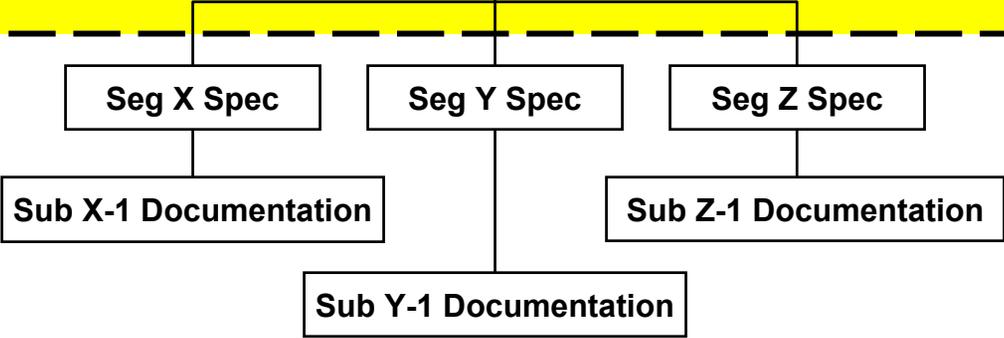
Operational
Requirement
Related



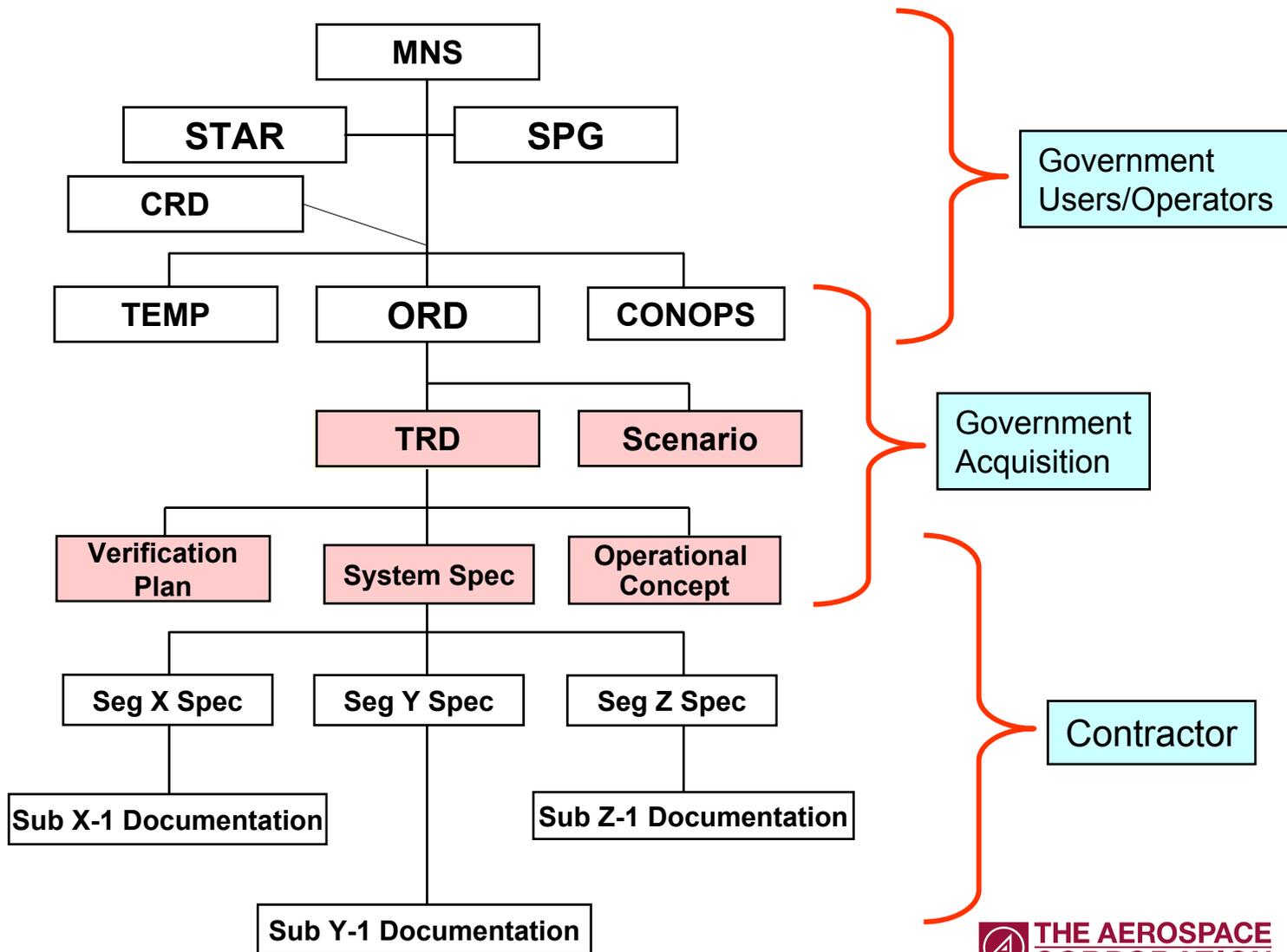
System
Requirements
Related



Allocated and
Product
Requirements
Related

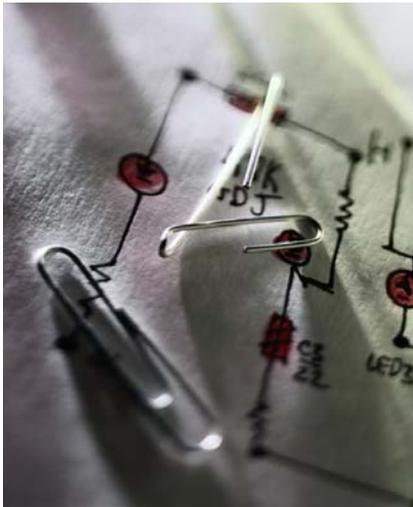


Government & Contractor Responsibilities



When is a Configuration Established or Changed?

- The Government system level configuration is defined as part of the program acquisition strategy
- The Contractor system-element-product configuration is defined in early production but matures with the program
- Changes to the baseline may or may not be system relevant
 - Government is a *partner* on system changes after System Specification approval



Class I (e.g. affects baseline; form, fit, function, cost, schedule, safety, ILS considerations, etc.)

Class II (e.g. documentation change).

Government personnel should review changes for correct classifications

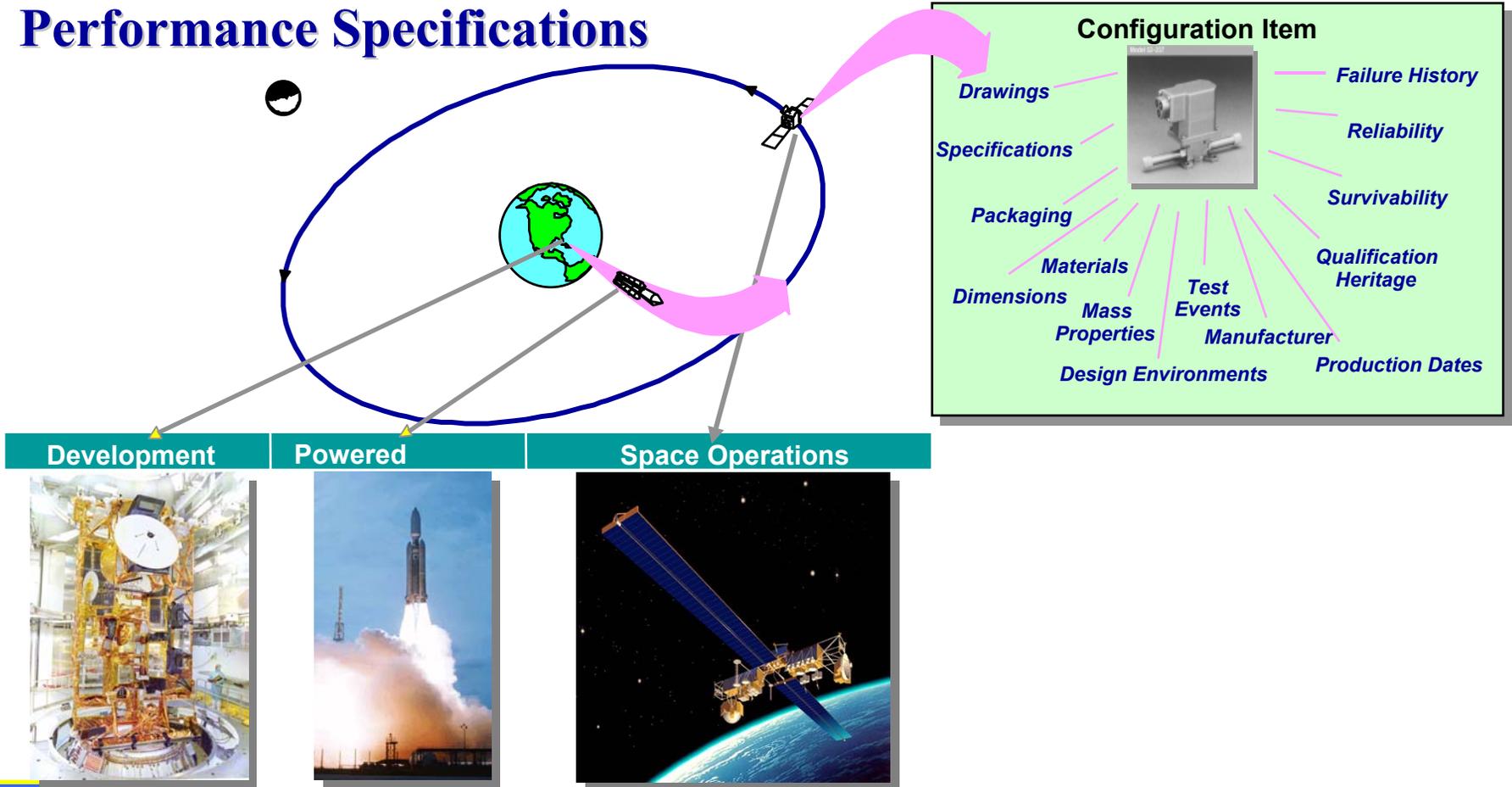
Photo Courtesy of The Aerospace Corporation



CM Assessment - Example

From a Systems View of Performance Specifications

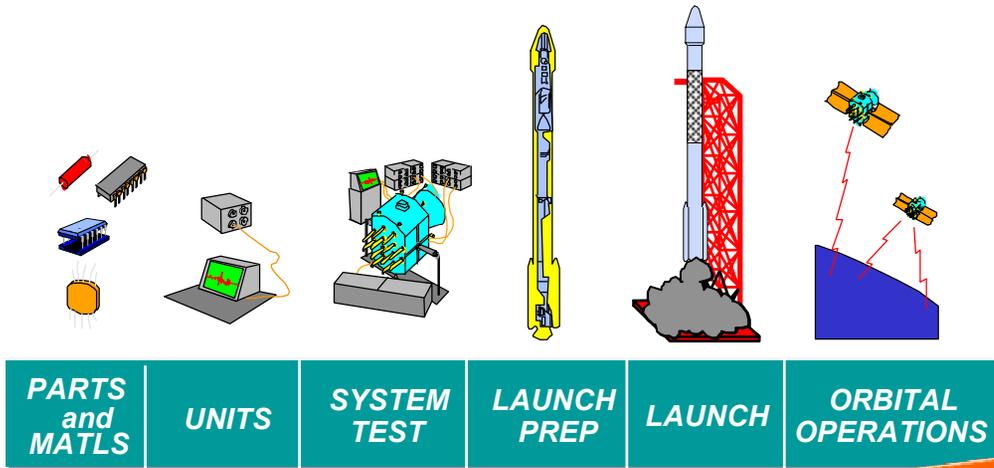
To the Factory Pedigree



Photos Courtesy of Lockheed Martin



CM Interface Lifecycle



“Verification Targets” :

- Documentation
- Processes
- Physical
- Functional
- Inspection
- PM&P

Multiple Subcontracts

“Complex Interfaces & Integration”

Design Feedback

Model Validation

Photos Courtesy of Lockheed Martin



Configuration/Change Management Mishaps

Program	Problem	Outcome
Interface Confusion		
Mars Climate Orbiter (1999)	Mix-up in metric/English units	Lost in space
Skipper (1995)	Russians used (+) polarity ground, US used (-) polarity	Battery drained
Titan IV B-32 (1999)	Misplaced decimal point	Milstar III put in wrong orbit
Poor Change Control		
Terriers (1999)	Software did not change to reflect the reorientation of a torquer coil	Solar array malfunctioned
SOHO (1998)	Command script revised without enabling gyro	Satellite lost control
Inadequate Requirements Management		
Mars Polar Lander (1999)	Part of a complex requirement statement did not flow down	Crashed on Mars
STRV (2000)	Software did not implement a pulse command requirement	Receivers burned out



Miscue Example – Titan CT-2 Failure

- Software engineers developed a specification for single-payload mission
- Hardware engineers redlined the draft specification to facilitate wiring, and designed the harness based on redlines
- Redlines fell through Mission spec's cracks – S/W and H/W incompatible
- Generic test masked problem
- Mission spec viewed as a software document and not subject to configuration control

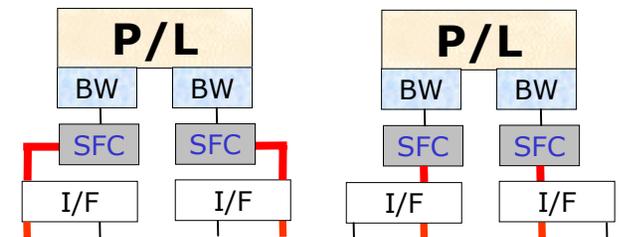
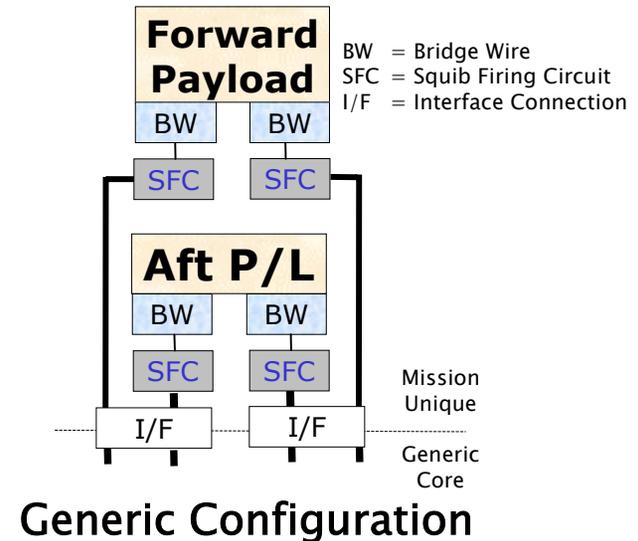


Figure Courtesy of The Aerospace Corporation



Configuration Management / Items to Remember

You should now understand

- What Configuration Management is & why it is necessary?
- How CM contributes to the Systems Engineering mission
- CM starts early and is critical through the contract lifecycle
- Gov't/Contractor roles in CM
- CM Must be worked by engineers, contracts and cost-account managers.... *"It takes as whole team"*
- Profound risk implications from a poorly structured or disciplined CM process

