NASA Armstrong Simulation Engineering

AFRC Core Simulation Overview

April 29, 2015
Manny Castro
Manuel.P.Castro@nasa.gov

Overview

Core Simulation

- Capabilities
- Configurations
- Simulation tools
- Simulation framework
- User interface controls
- Simulation GUI
- Core framework
- Model and data preferred formats
- Core simulation environment
- Software Engineering Processes
- RAIF facility

Simulation Capabilities

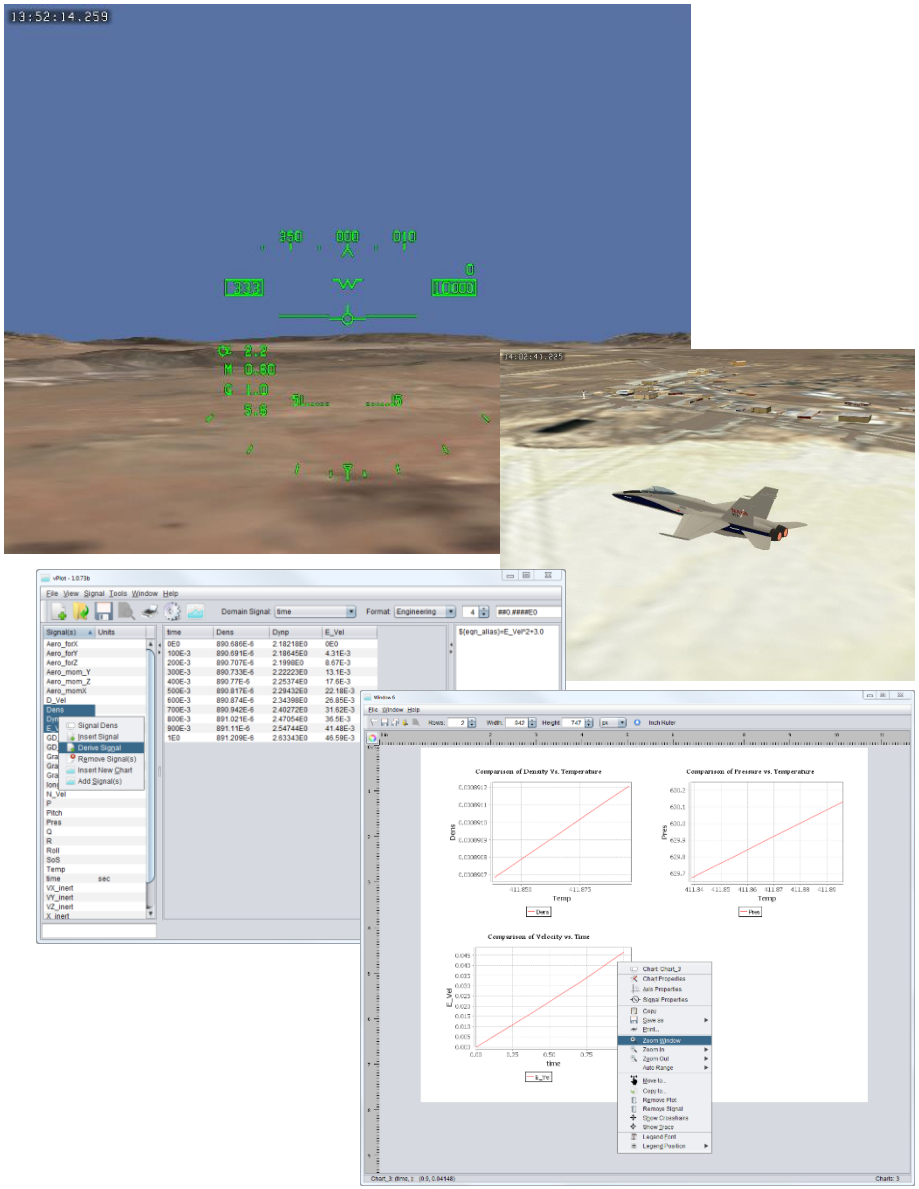
- **Fix-based engineering simulations**
  - Non-linear
  - 6 Degrees-of-freedom
- **Used for design and flight performance evaluations**
- **Operable by one person**
- **Interfacing with flight hardware is routine**
  - Mil-STD-1553
  - ARINC 429
  - Ethernet
  - RS-232 / RS-422
  - Analog and Discrete I/O
  - PCM streams

Simulation Configurations

- **Same software supports:**
  - Non-real-time (batch)
    - Desktop
  - Real-time, interactive mode
    - Cockpit
    - 3-D Visuals
    - Execution rate tied to clock
  - Hardware-in-the-loop
    - Mission Computers
    - Flight Control Computers
    - Actuators
    - Health Monitors
    - INS/GPS
  - Vehicle-in-the-loop

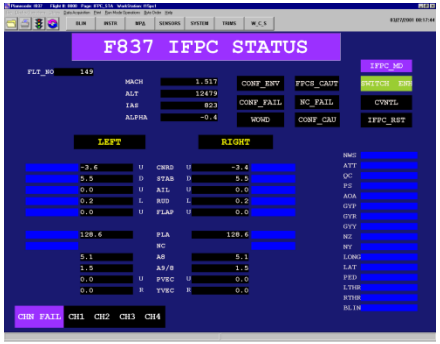
Simulation Tools

- **Real-Time 3-D graphics**
  - Multiple views available
  - Multiple aircraft in same virtual space
  - Articulated surfaces, smoke trails, etc.
  - Heads-up Display (HUD) symbology
  - Path-in-the-sky (Pilot pathway guide)
- **DTH – Dryden Time History**
  - Data conversion and analysis
- **Quick Plot (**Plotting program**)**
- **NuPlot (**Plotting program**)**
- **TCP I/O (**Matlab/Simulink I/F**)**
- **TALK (**Remote control of sims**)**
- **McGUI (**Develop Monte Carlo scripts**)**
- **Google Earth (**moving map**)**
- **MATLAB/SIMULINK Autocoder**

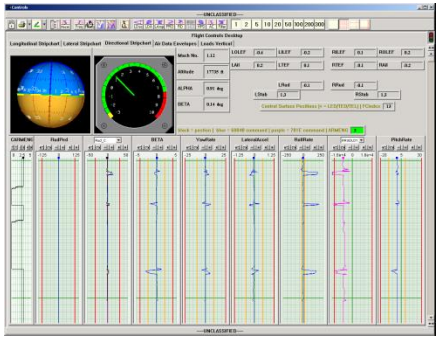


Simulation Tools (Cont)

- **Mission Control Room displays**
  - Can drive control room displays directly from the simulation (PAM3D, PDS, IADS)
  - Engineers can prototype displays
  - Engineers train while testing



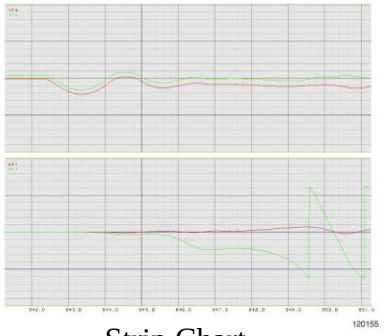
Control Room PDS Display



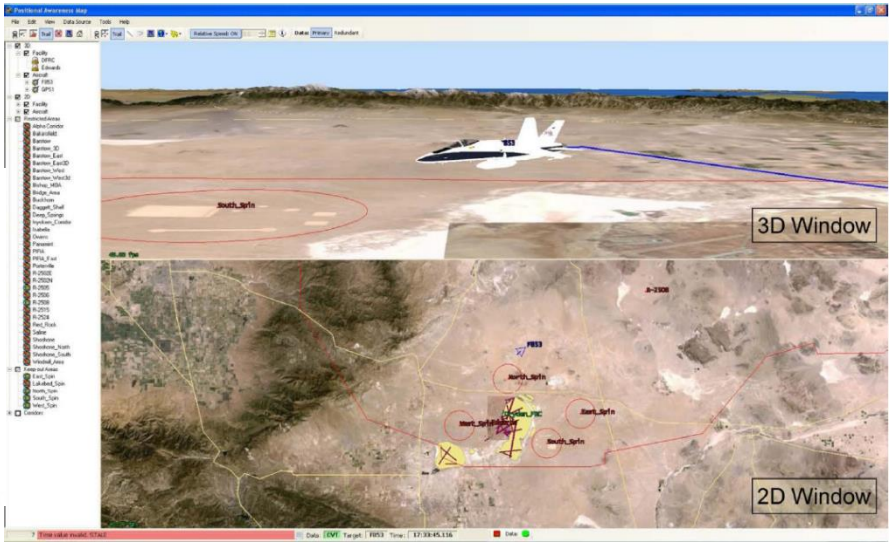
Control Room IADS Display

- **Telemetry encoding (PCM)**
- **Real-Time data recording**
- **Heads Down Display (HDD)**
- **Strip Chart displays**
- **Real-Time data playback**
  - Flight data
  - Previous simulation run

Heads Down Display



Strip Chart Display



PAM3D

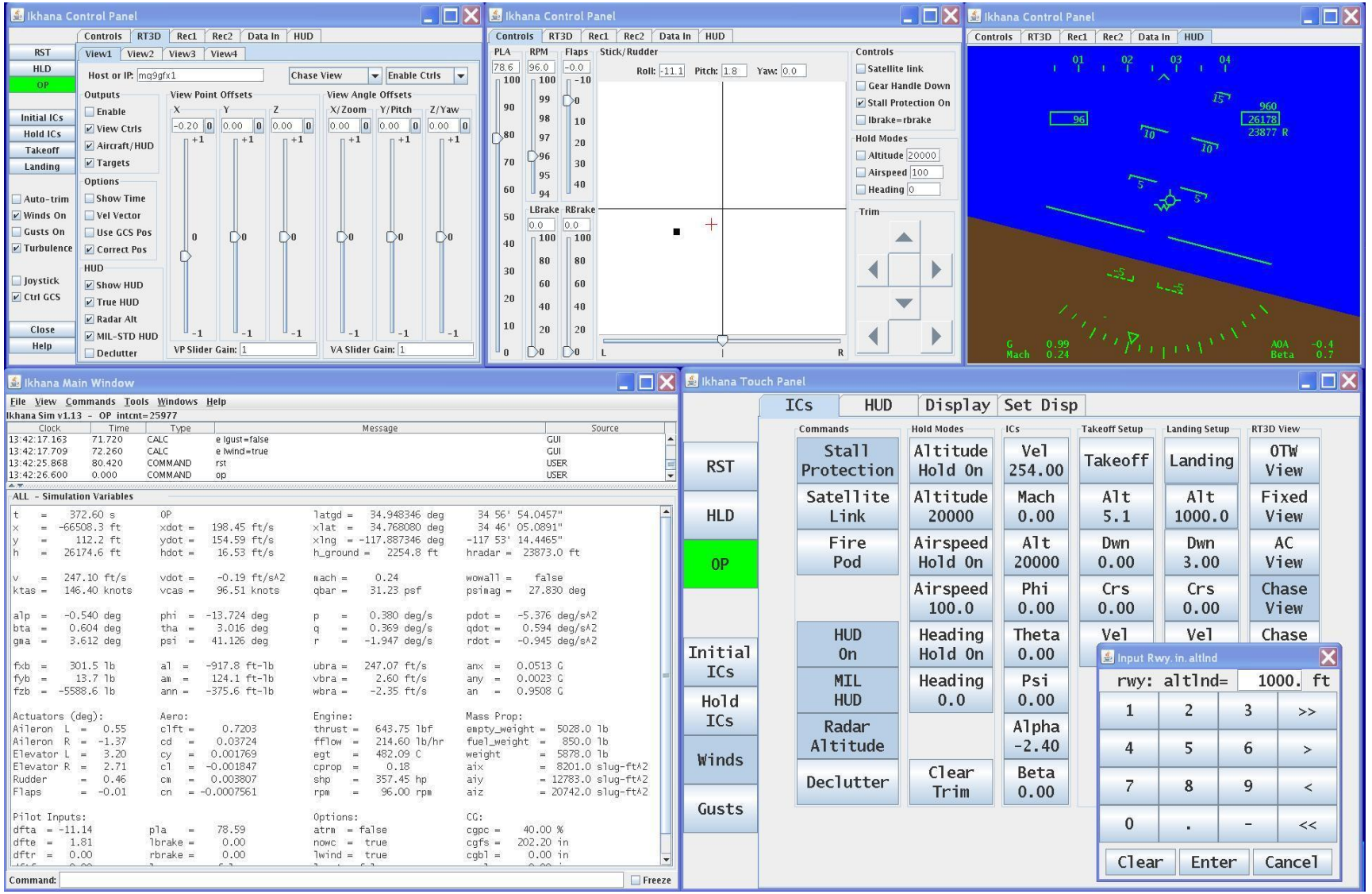
Simulation Framework

- **Armstrong Coresim Version 7**
  - C++, JAVA, Interfaces to FORTRAN source
- **Coresim Framework Components**
  - Equations of Motion
  - Atmosphere/Winds
  - Terrain
  - Real-time Control
  - Graphical User Interface
  - Command Processor
  - Scripting Language
  - Event Logging
  - Code Generators
  - Automated Testing
  - Data Recording/Playback
  - Linear Model Generator
  - Telemetry Encoder
  - Hardware Interfaces
  - External Application Interfaces
  - Help Pages
- **Vehicle-specific Models**
  - Vehicle models of subsystems
  - Vehicle-specific I/O interfaces
  - Actuator models
  - Aerodynamics model
  - Mass Property
  - Trim
  - Linearizer
  - Gear model
  - Engine models
  - Control System
  - Cockpit

User Interface Controls

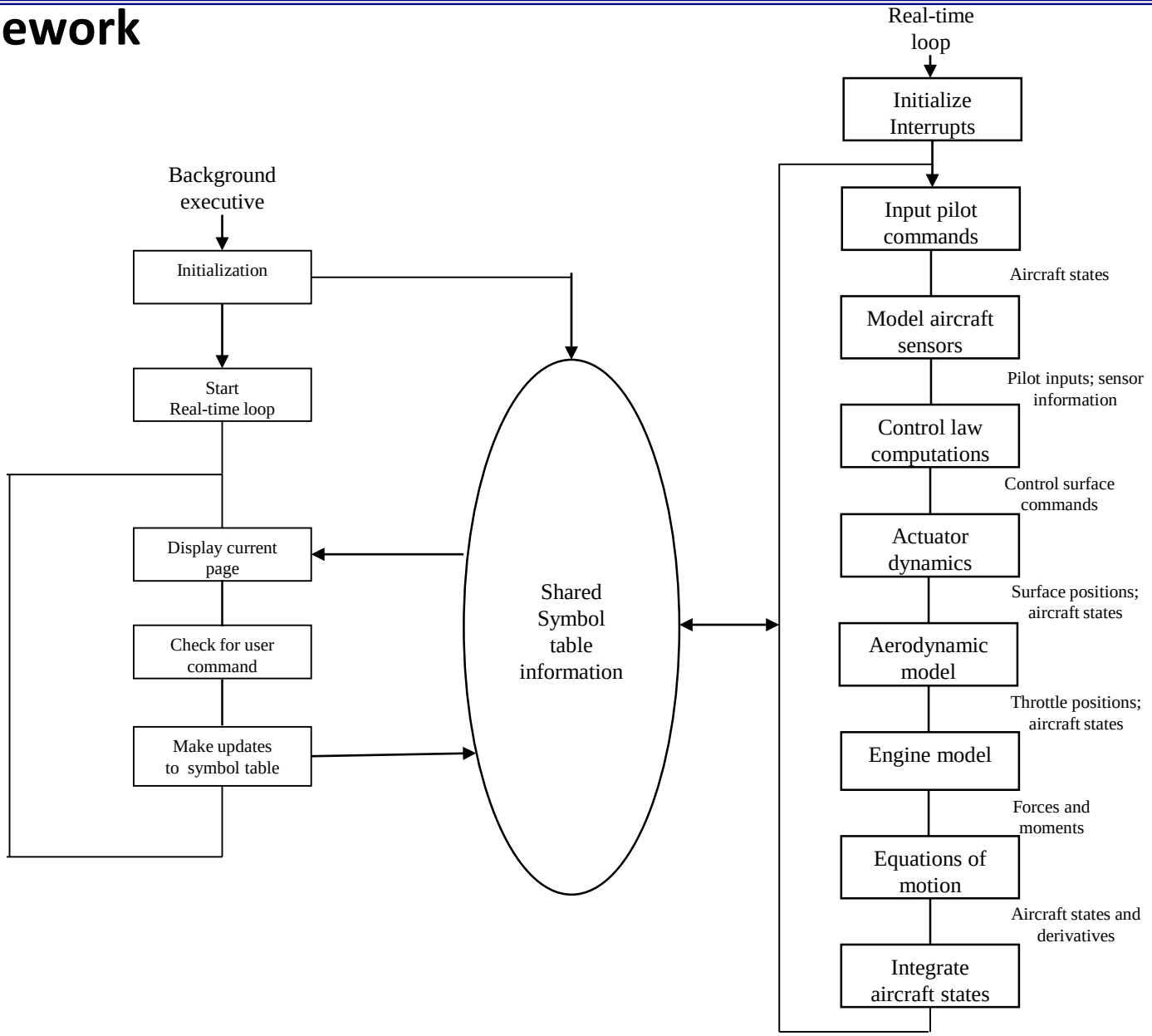
- Graphical User's Interface (GUI) displays are programmed in Java for portability.
- Simulation is command driven.
  - All commands, including those on the GUI, can be scripted.
- Display pages show parameters of interest.
- User can customize and save window layout.
- Tests can be automated to ensure repeatability and quick turnaround time.
- Faults can be injected to test off-nominal conditions.
- Pre-defined interface to external applications (e.g. Matlab, Simulink, D-Six, graphics software, Ada flight code).

Simulation GUI



Core Framework

- **Framework**



Model and Data Preferred Formats

- **Data Dictionary**
  - Plain-text free format files with syntax similar to C programming language, *.def files
  - Files read at start time to produce an object-oriented symbol table for runtime access

```
double fy_aero {
        label = "Y body axis aerodynamic force at aero reference";
        units = lbs;
        sign = +right;
        default = 0.0;
        }
```

- **Configuration Files**
  - Plain-text free format files with syntax similar to C programming language, *.cfg files
  - Files are read and parsed during initialization without need to recompile

Core Simulation Environment

- **UNIX**
  - Linux
    - Redhat 6.5
    - Oracle UBE Linux 6.5
    - CentOS6.5
    - GNU Compilers and tools
  - Oracle Solaris
    - Solaris 10
    - Solaris 11 X86
  - Linux real-time
    - https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_MRG/2/pdf/Realtime_Tuning_Guide/Red_Hat_Enterprise_MRG-2-Realtime_Tuning_Guide-en-US.pdf
    - http://h10032.www1.hp.com/ctg/Manual/c01804533.pdf
    - OS is configured for CPU isolation
    - Application (Core Framework) has additional configurations
- **Computer Hardware**
  - Small scale computer for low levels of I/O
  - Large multi-core computers for multiple I/O devices
  - Dell T1700 – Oracle X4-4 compute platforms
  - XEON Processors

Software Engineering Processes

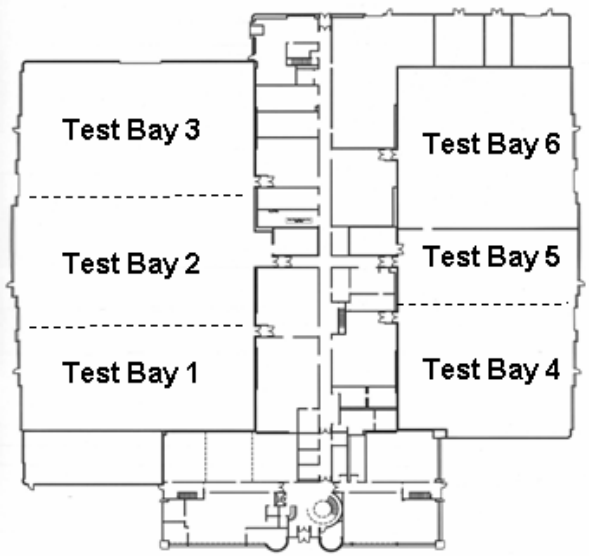
- **NPR-7150 (NASA Procedural Requirements)**
  - Core 7 Simulation Framework is NPR-7150 Class D compliant and DPR-7150 Class III compliant
  - CORE Configuration Control Board (CCB)
    - DR, CR, STR tracked in CCB centers database
  - Software Users Guide
  - Programmers Guide
  - Unit test
  - Automated System test using Core framework scripting checkcase capabilities
  - Version tracking and storage with SVN repository
- **Core 7 Simulation Framework**
  - Baseline Software for starting a new project
  - New features and bug fixes are incorporated
- **Core 7 Simulation Production**
  - Follows project's engineering processes
    - Based on NPR-7150 and DPR-7150
    - Project CCB
  - Models are obtained from external sources
    - Aero Branch
    - Controls Branch
    - External customers
      - CFD, Wind Tunnel, or other sources of model data
  - Models are validated with checkcases obtain from model source and data from actual flight test

Research Aircraft Integration Facility

The **RAIF** and its simulation capability provide project teams with the means to conduct efficient and thorough testing of advanced, highly integrated vehicles.

The RAIF was specifically designed to support the development, integration, test and validation of highly integrated, complex research vehicles.

- Six vehicle test bays including vehicle support systems (e.g. vehicle cooling and power)
- Eleven simulation labs, avionics and mechanics shop, ESD qualified areas
- Simulation labs overlook the test bays and are connected to the bays for data, video and audio communication
- Capable of remote testing and monitoring
- Office space and conference rooms available



Existing Facility Layout