



**U.S. AIR FORCE**

# Building Distributed Simulations Utilizing the EAAGLES Framework

**Douglas Hodson**

ASC/XR – SIMAF (WPAFB)

[doug@OpenEagles.org](mailto:doug@OpenEagles.org)



# Topics

- **Frameworks, Toolkits & Applications**
- **Specifics & Real-Time Simulation**
- **Framework Organization**
- **Simulation Design Pattern**
- **Graphics Architecture**
- **Sample Applications**
- **OpenEagles**

# Frameworks, Toolkits & Applications

- **Terminology**
  - **Framework:** a set of cooperating classes that make up a reusable design for a specific class of software.
  - **Toolkit:** a set of related and reusable classes that provide useful, general-purpose functionality
    - Examples: OpenGL, OpenSceneGraph
  - **Application:** are stand-alone executable software programs. They typically satisfy a particular need.
    - Examples: Microsoft Word, Quicken, Fighter Cockpit, MQ-9 Ground Control Station
- **Frameworks make a lot of sense in the area of simulation, especially distributed where much of the code can be reused.**

# What is EAAGLES?

- **EAAGLES = Extensible Architecture for the Analysis and Generation of Linked Simulations**
- **A simulation framework that is used to design robust, scalable, virtual, constructive, stand-alone, and distributed simulation applications.**
- **It leverages modern object-oriented software design principles while incorporating fundamental real-time system design techniques to meet human interaction requirements.**
- **It provides abstract representations of system components so that multiple levels of fidelity can be easily intermixed and selected for optimal runtime performance.**

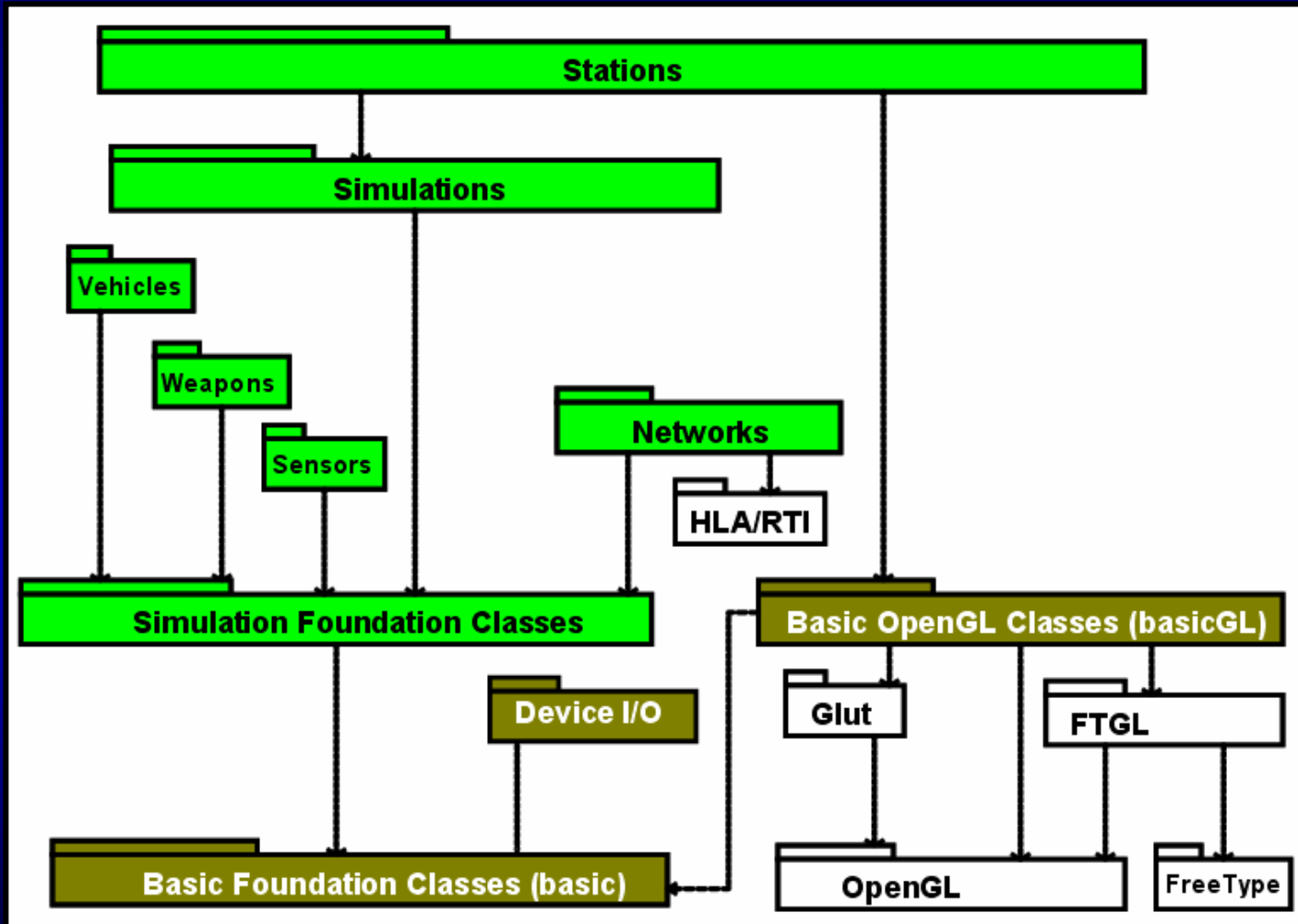
# Detailed Specs

- **Language: Object-Oriented C++**
- **Platform: X-Platform, Windows/Linux/Unix**
- **Cyclic-based scheduler (foreground/background)**
  - Basic real-time system design concepts interlaced throughout class hierarchy
- **Serves as a simulation design pattern.**
- **Provides abstract network interface to support DIS, HLA and other interoperability protocols.**
- **Graphics toolkit that enables the rapid construction of operator interfaces**
  - Strong adherence to model-view-controller paradigm
- **Easy to parse, easy to edit/read, Scheme-like input language**
- **Framework purposely does not include a main() function.**
  - Developer controls the application, not the framework
  - Developer manages threads
- **Executable applications built off the framework**

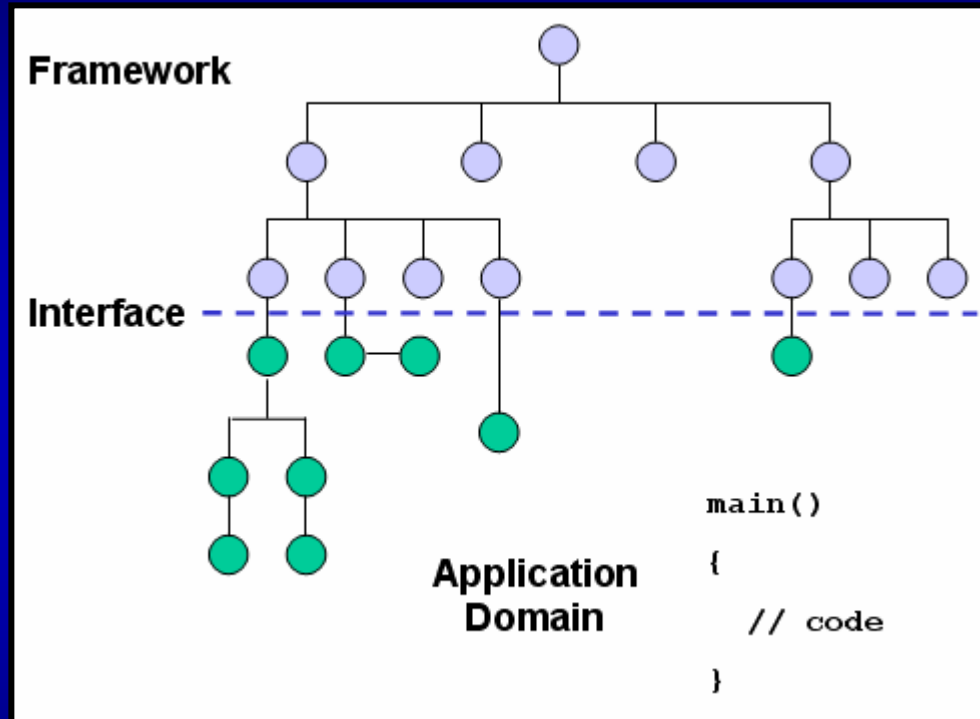
# A Framework that Embraces Design Constructs from Real-Time Systems

- **Simulations that interact with human participants must respond within a prescribed deadline (latency or response time).**
- **How to organize code to meet this requirement falls into the category of a real-time system.**
- **A typical structure for a real-time system is a cyclic-based scheduler or commonly called a foreground/background system.**
- **EAAGLES is a cycle-based (frame-based) system, NOT a discrete event based system**
  - **Better for supporting interactive simulation requirements**
  - **Higher fidelity system models typically need to run at some frequency in order to operate correctly**

# Package Organization



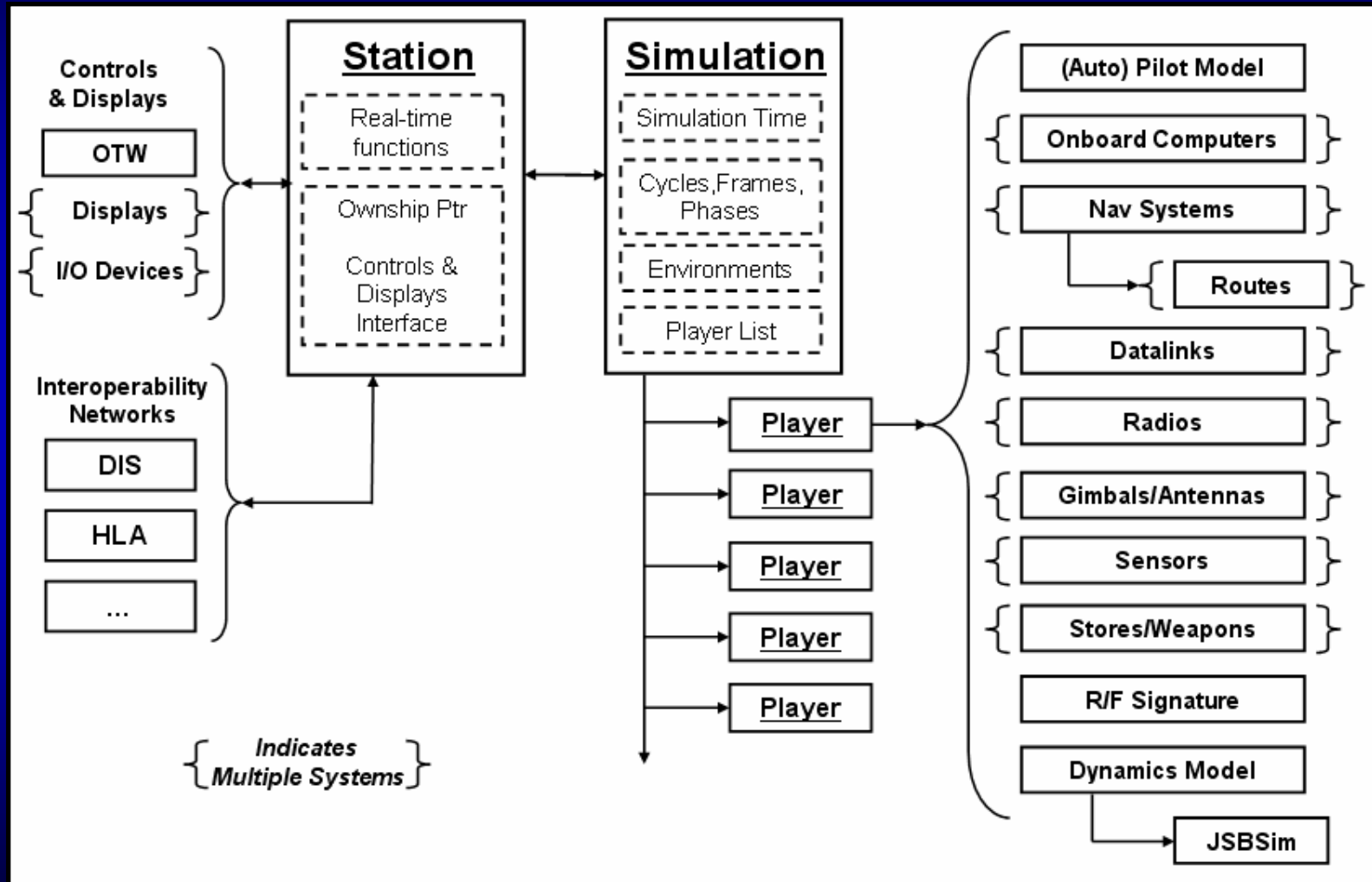
# Application Building and Component Tree



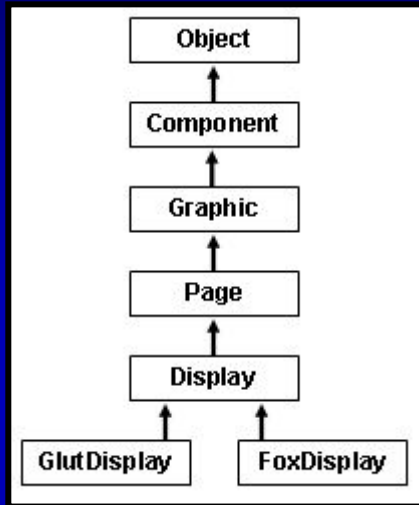
- Software is partitioned to support foreground and background code execution. The equivalent of a time-critical thread is interwoven into the class hierarchy to explicitly support virtual simulation requirements.



# Simulation Design Pattern



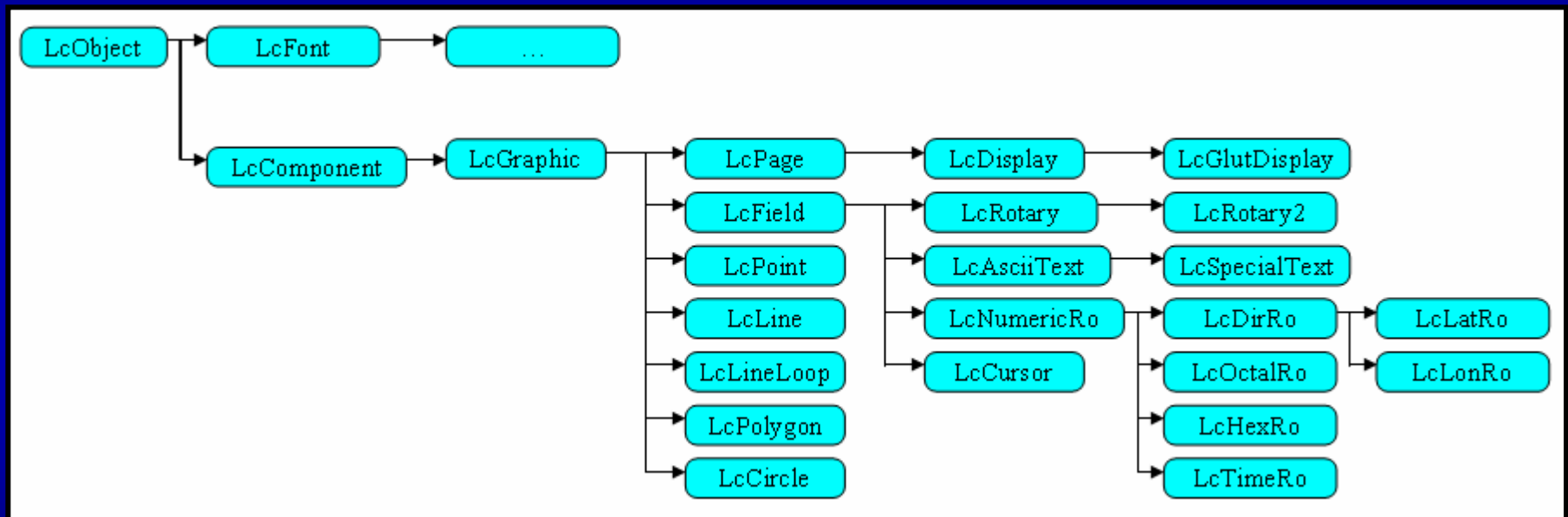
# Graphics Architecture



- Designed for interface displays
- Rely on open source toolkits for building GUIs (Fox/FLTK)



# Sampling of basicGL Classes



# Fighter Cockpit



- The Fighter application was built utilizing the EAAGLES framework. The heads down display was built by extending the provided graphics toolkit.
- The heads up display is drawn by the SubrScene application.

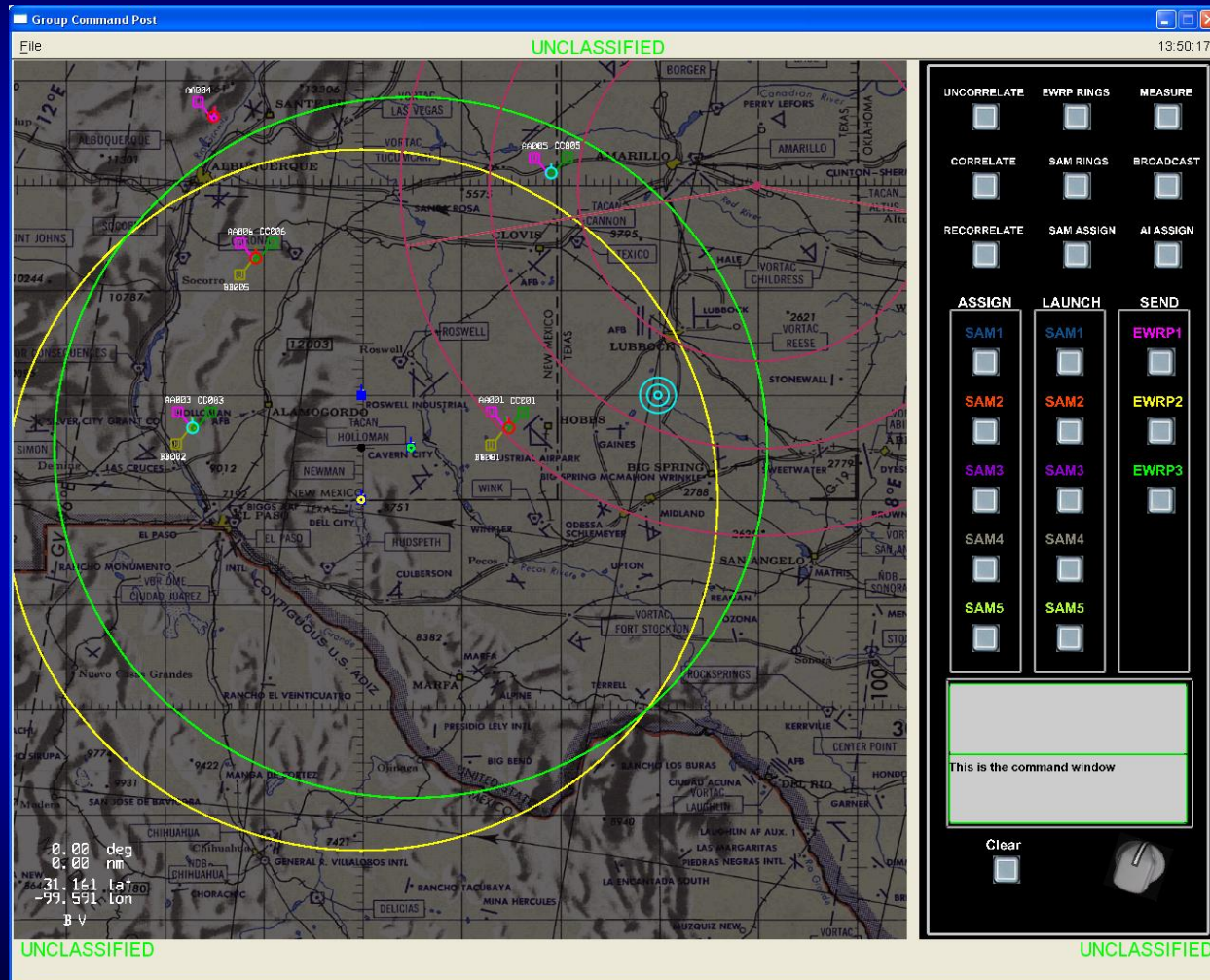


# MQ-9 Ground Control Station



- EAAGLES is the foundation for several applications.
- The Fox open source GUI toolkit is leveraged with the EAAGLES graphics toolkit to provide operator interfaces.
- Subscene is utilized for drawing camera views as seen by the UAV.

# Group Command Post (GCP)

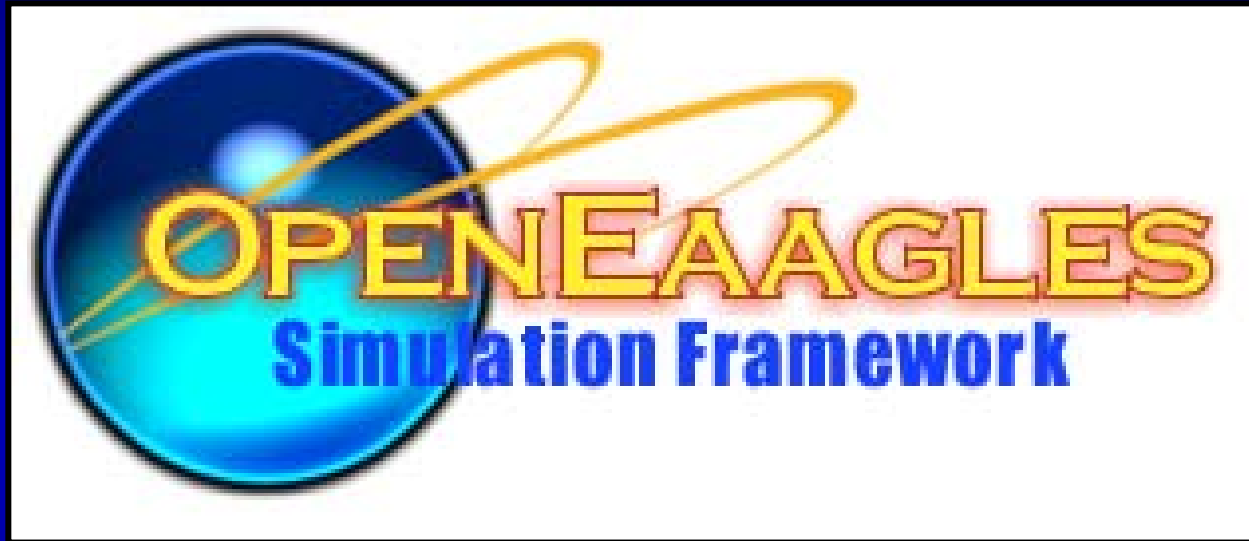


- This application along with two other EAAGLES-based applications form the core of an IADS infrastructure.
- The FLTK open source GUI toolkit is leveraged with the EAAGLES graphics toolkit to provide the operator interface.

# Final Thoughts, Almost

- **The EAAGLES framework is designed for the simulation application developer; it is not an application itself.**
  - **Applications are built utilizing the framework.**
  - **Strict adherence to the model-view paradigm is observed.**
    - **Connecting models to views is done at a higher level.**
    - **An application developer can choose to leverage simulation or graphics capabilities independently if desired.**
- **The framework embraces the object-oriented paradigm and therefore system abstractions while interweaving design concepts from real-time systems.**
- **Government owned and managed, collaboratively developed software.**
- **In order to encourage the use of the EAAGLES framework throughout the community, a nearly fully featured version has been released into the public domain.**

# Public Released Version



- Can be downloaded from [www.OpenEaagles.org](http://www.OpenEaagles.org)
- Closely tracks enhancements to EAAGLES, but does omit some functionality.