CORBA in Control Systems

Presented at the OMG Workshop On Distributed Object Computing For Real-Time and Embedded Systems, Washington, DC, 14-17 July 2003

Dr. Shahzad Aslam-Mir, Prism Technologies

Dr. James L. Paunicka, Boeing

Eric J. Martens, Boeing

- Sam's slides:
  - Intro
  - Control Systems
  - Importance of CORBA in Control
  - Current schools of thought on CORBA applicability in control systems
  - Historical perceived limitations on CORBA use in control loops
  - Case studies

Flight Control Example

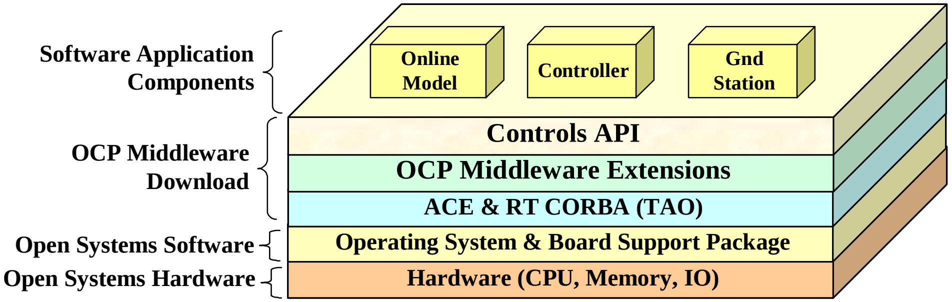
- Standards-based CORBA middleware used in live flight of a commercial rotorcraft UAV in May 2002
  - Flight test was collaboration between Boeing and Georgia Tech School of Aerospace Engineering
    - » Boeing – embedded software architecture, integration of software on embedded processing platform
    - » Georgia Tech – flight vehicle, vehicle control software

Flight Control Example

- Flight test was part of the DARPA IXO (Information eXploitation Office) SEC (Software Enabled Control) program
  - Significant technical direction from Air Force Research Laboratory
    - » Information Systems, Advanced Architecture and Integration organization
    - » Air Vehicles, Control Systems Development and Applications organization
  - Leverages OCP (Open Control Platform) being developed on the SEC program
    - » Middleware platform that adapts Boeing Bold Stroke software technology to the domain of flight vehicle control
    - » ACE/TAO
    - » Run-time optimizations to support flight vehicle control
    - » API for flight vehicle control applications (“Controls API”)
    - » Developed by Boeing-led team that also includes Georgia Tech, University of California-Berkeley, and Honeywell

Background

- SEC funds two technology areas
  - Control technology for flight vehicles
    - » Fixed-wing and rotorwing UAVs
    - » Multiple research teams from industry and academia
  - Enabling Software technology
    - » Adapt Boeing Bold Stroke software technology to the domain of flight vehicle control



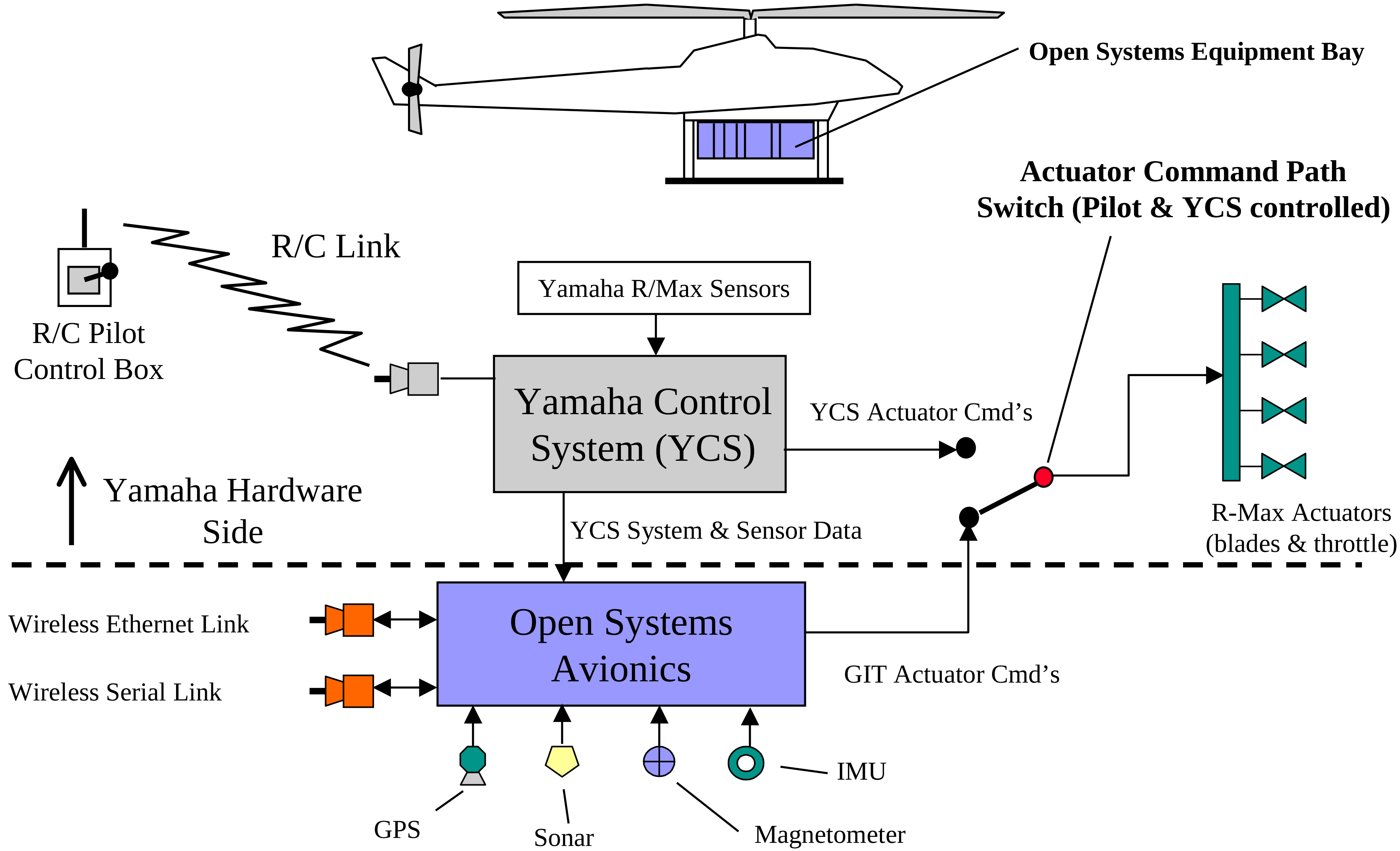
Commercial UAV Flight Vehicle

- Yamaha R-Max autonomous helicopter
  - Couple hundred pounds,
  - 10-foot main rotor diameter
- Fitted with open systems avionics platform for SEC program experimentation
  - Sensors
    - IMU (Inertial Measurement Unit), GPS, sonar for altimeter, magnetometer for compass
  - Actuators
    - Throttle, main rotor, tail rotor
  - Comms
    - Wireless ethernet
    - Wireless serial link
  - Onboard compute platform
    - Single 266-MHz Pentium II processor

R-Max Vehicle

BOEING

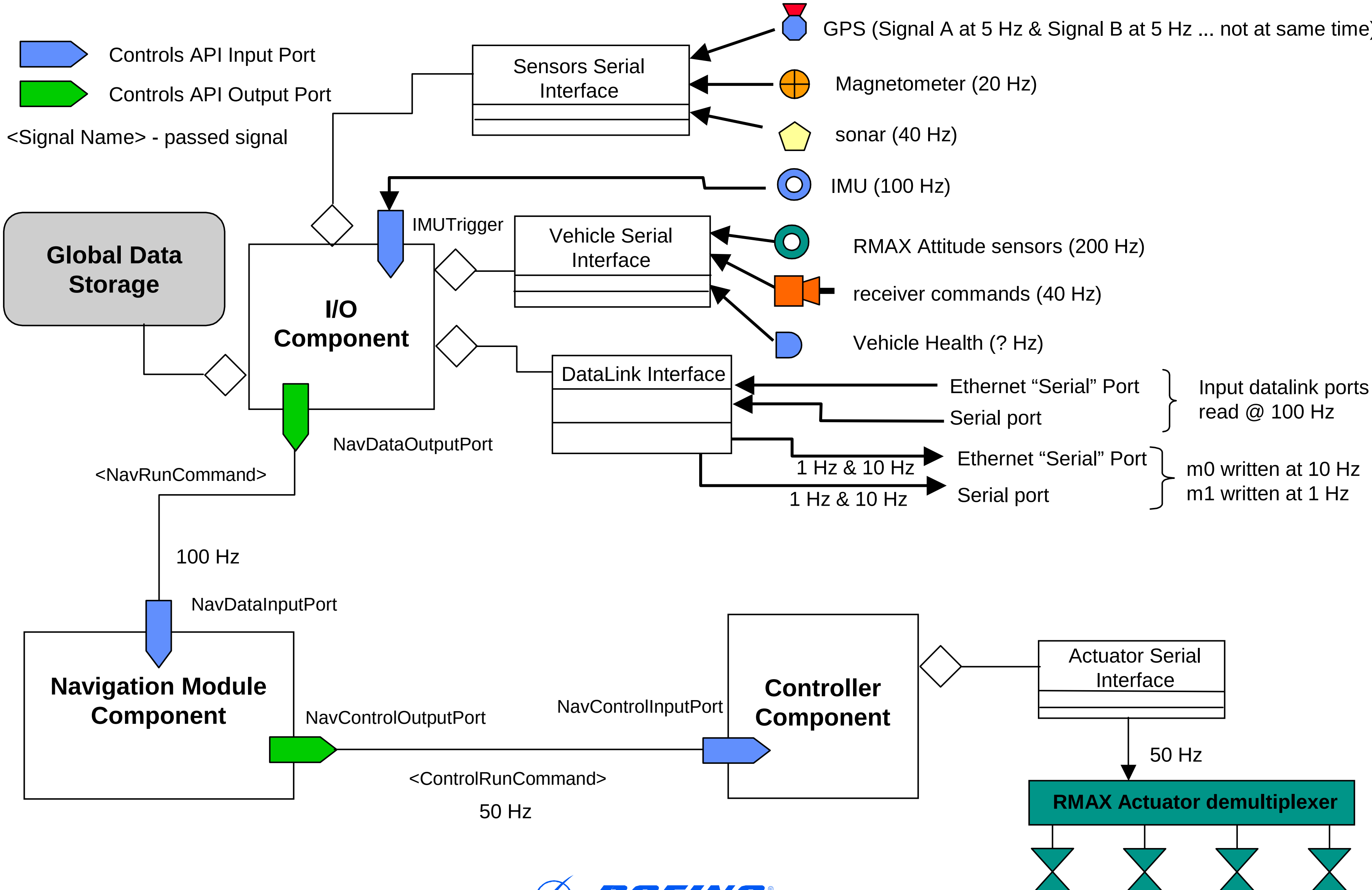
Autonomous Helicopter Electronics



Flight Test Embedded Software Architecture

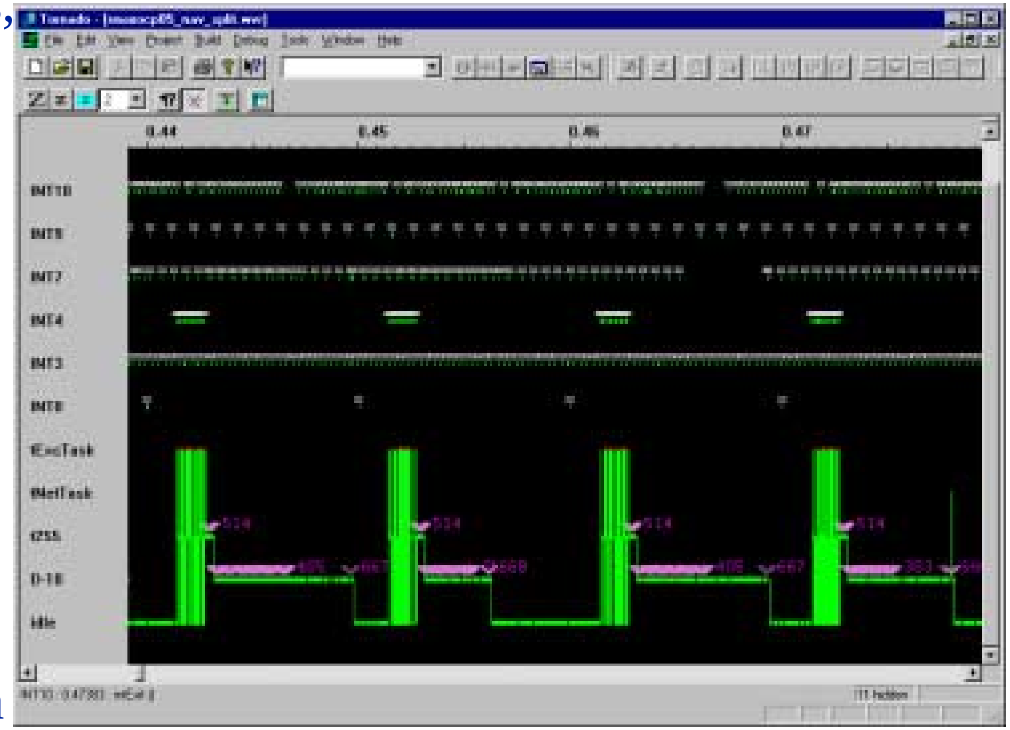
- Major Elements
  - Lowest level – VxWorks RTOS and appropriate BSP (Board Support Package)
  - Middleware level – Open Control Platform from SEC program
  - Application software level – multiple components written by Georgia Tech
- Run-time configuration of CORBA-based software
  - Middleware triggered execution of multiple software components with EC (Event Channel)
    - » 100 Hz operation
      - Triggered the start of inner-loop control processing with the arrival of IMU data
  - Middleware mediated I/O among the various aircraft sensors, flight control actuators, and multi-level control loops
  - Implemented a software reconfiguration in flight
    - » Neural net adaptive controller switching to a conventional inverting controller

Flight Test Embedded Software Architecture in Context



Flight Test Embedded Software

- Multiple Application Components, including
  - I/O Handler
  - Navigation Processing
    - » Also handles data link updates
  - Controller Processing
- OCP Frame Manager launches 100-Hz loop after 16-byte IMU burst
  - Pushes an event to start I/O Handler
- Other components also initiated in 100-Hz frame with event pushes
- Middleware-Based Reconfiguration accomplished with OCP activating and deactivating different controllers



WindView Plot of OCP-Based Application

Flight Test Timeline

- R/C (Radio Control) pilot performed take-off (and subsequent landing) using baseline Yamaha flight control system
- While in flight, rotorcraft switched to open systems research flight control system
  - With RT-CORBA-based flight control program
- Transition between vehicle controllers (neural net to conventional inverting controller) triggered by ground station command and accomplished by middleware

Video from Flight Test

- Sam’s slides:
  - Analysis of where we are
  - Conclusions