EVLA Software

Communications Infrastructure

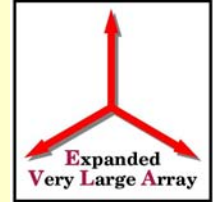# EVLA Communications
Design Principles

EVLA Communications Design Principles

- ## Communicate Less

  - ### Using more autonomous components,

    - The more functionally self-contained a module is, the less that must be communicated to it:

    - Such modules exhibit *tight cohesion* and *loose coupling*.

      - cohesion: functional 'single-mindedness'

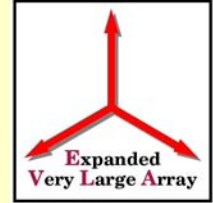   - coupling: interaction necessary to perform its job

EVLA Communications Design Principles

- # Communicate Less (cont.)
  - ## Using Hierarchical Layering to Communicate only Relevant Information
    - ### Information is more general at higher layers, more specific at lower layers.
      - Proposal layer refers to an observed source by its sky position.
      - MIB as a set of bits in a servo position register.
    - ### Irrelevant information need not be communicated
      - There is no need for Proposal layer to see MIB bits

EVLA Communications Design Principles

- # Communicate Less (cont.)
  - ## Using Hierarchical Layering (cont.)
    - ### Number of Communications Channels Maintained by any one Component is Reduced
      - Instead of Executor directly communicating with all 900 MIBs, it communicates with 27 antennas that each communicate with 30 MIBs.
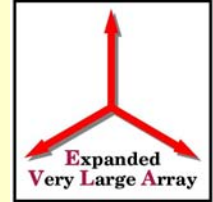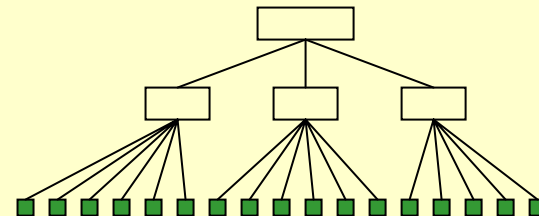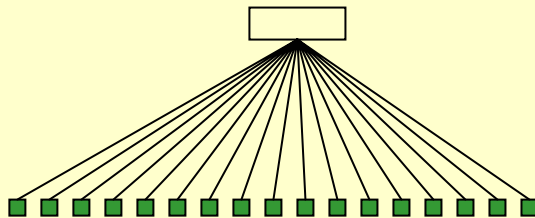
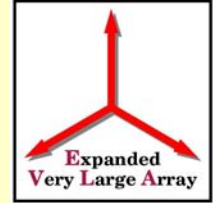# Scope of the
EVLA Communications System

Scope

- ## WHAT - it is:
  - A means of passing information between the components within the EVLA system and between the EVLA system and its users.

- ## WHAT - it Provides:
  - Control/Monitor Communications
  - Alert Reporting
 - Archive (Monitor Data/Alerts - not correlator output)

Scope

- **WHO - Uses it (Functionally)**
  - Components within the EVLA System
    - Scheduler, Executor, M & C, etc.
    - The normal Sized Lines on the big Data Flow Diagram
  - Users of the EVLA System
    - UI's and user applications
      - Operator consoles
      - Data logger/analyzer applications
    - Tech/maintenance applications
      - might plug a PDA directly into a MIB in the antenna or on the test bench
      - Offline maintenance programs
    - Internet Users
      - Astronomer's observing status
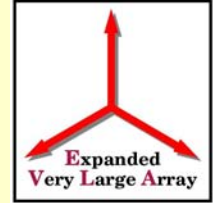   - Web 'lurkers'

Scope
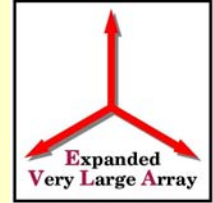
- **WHO - Uses it (Physically):**
  - CMP
    - 68040 / VxWorks / 'C'
    - PPC / Linux
  - MIBS
    - TriCore TC11IB / Nucleus OS / 'C'
  - EVLA System Computers
    - mchost / Linux (not resource limited)
  - Misc. Desktops / Laptops (i.e. not resource limited)
    - Unix/Linux/OSX, Windows, ?
  - PDA's / Handhelds
    - PalmOS / Linux / ?
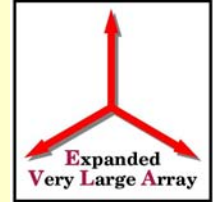
Scope

- WHERE - The Users are Located:
  - At the Site
    - Control Building & Other buildings at the site
    - In an Antenna (tech doing maintenance)
  - AOC & other NRAO networked places
    - In an Office
    - On a Test Bench in a Lab
  - At Home
    - When the Dreaded 3:00 AM Phone Call Happens
  - Across the World
    - Web Access to Portions of the System
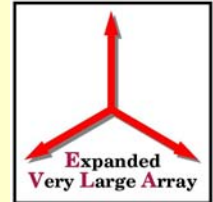
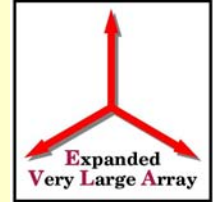# System Description
and
Constraints

Physical Description

- IP/Ethernet
- 1000/100 Mbit/s
  - MIBS & CMP are 100 Mbit
- Fiber Optic (almost everywhere)
  - Single mode between antennas & Control Building
  - Multi mode elsewhere (including in antenna)
- 10.80.1xx.yyy
  - xx - major component (antenna, Control Building)
  - yyy - sub-component (MIBs, weather station, mchost)

Constraints

- IP/Ethernet is not a Real-Time Network

- MIBs are Resource Limited

  – Large COTs middleware will not physically fit

- Traffic Congestion

  – Though the overall bandwidth of the system will accommodate communications, the large number of processors (some 900 MIBs alone), and the synchronous nature of the system may cause 'spikes' at certain times.
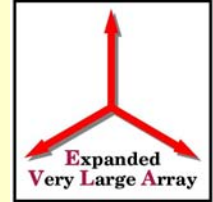
Inter-process Communications

or

How the Various EVLA Subsystem Processes
Communicate with each other During Normal
System Operation

EVLA Inter-process Communications

- The EVLA is Hierarchical in Operation
  - Scheduler instructs Executor which instructs antennas which instruct MIBs (EVLA) or Datasets (VLA)
- Generally - Control Data Flows Down, Monitor Up
- Control Information is Passed via Messages
  - Function calling is not attempted across the network because of its real-time constraints
  - All functionality for executing a command is contained locally
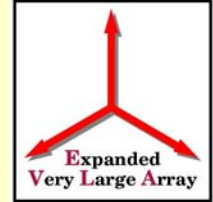
EVLA Inter-process Communications
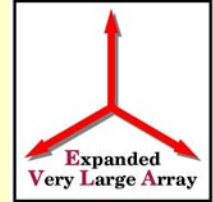
- ## Components are not 'Programmed' to Send Monitor Data*
  - There is no code in the Wx Station to send 'temperature' to Scheduler every 10 minutes.

- ## Instead, Components Behave as 'Monitor Point' Servers
 - Scheduler simply fetches the required information from the weather station whenever it is needed.

*   The monitor data archive (where every monitor point is sent periodically) is not considered 'inter-process communications'.

EVLA Inter-process Communications

- Most Monitor Points are not Communicated
  - **The vast majority of monitor data is irrelevant to inter-process operation**.

- Only 'Front panel' type information is sent
  - Things that are useful to *operation*
    - like general status, locked on frequency, servo position, switch setting
    - Not things like internal power supply readings
      - These are generally irrelevant to operation from a higher layer's perspective.
    - Scheduler wants to know if a particular antenna is working or not and is not interested in why or why not.
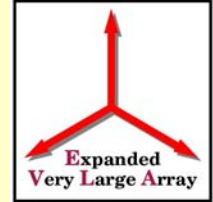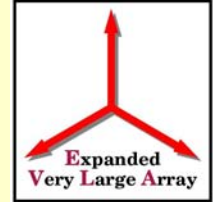
Alert Reporting

## Is Similar to Monitor Data Operations

Alert Reporting

- Alerts Communicate Exceptions to Normal Operation
- Alerts are Reported in Two Ways:
  - Every Alert will be Archived
  - Clients Subscribe to a Server's Alerts
- Types of Alerts will Include:
  - Failure - No longer functional; 5V power supply 0V
  - Warning - Out of Tolerance; 5V PS reads 4.5V
  - Error - Software/Human Related; divide-by-zero, illegal mode
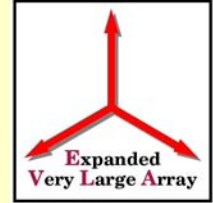 - Information - Information related … er … information

Alert Reporting
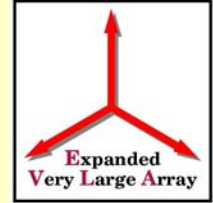
- Information About each Alert will Include:
  - Time - that it happened
  - Device ID - of the reporting device (not necessarily the root cause)
  - Type - of alert
  - Description - of the alert
  - Consequences - impact to the system or operation
  - Cause - to the level known
  - Action - suggestion to operator ('call Wayne', 'take offline')
- Static Info will not be Sent with Alerts
  - Descriptions and the like will be available as a hyperlink to a web page or other database type lookup

Alert Reporting

- Too Many Alerts can be Overwhelming
  - Important Information can become 'Lost in the Noise'

- Ways to Reduce Alert Noise:
  - Reduce 'flicker' where a value passes rapidly in and out of tolerance - e.g. apply hysteresis to the in/out-of-range values
  - Send only alerts that are asked for
    - During operation, clients subscribe to a server's alerts
    - The client only sees the alerts that he asked to see, no others
  - Use Hierarchy to Receive only relevant alerts
    - A power supply failure is not meaningful to Scheduler
      - whether or not an antenna is working is meaningful

Summary

- A team has been created to investigate various communications technologies.

    – Systems such as ACS will be implemented on a prototyping test bed.

    – The goal is to produce sufficient metrics about each system to facilitate selection (and to document justification) by approx. Q4, 2004.

Summary

Our goal is to build a system whose operation is based on *policy* rather than *constraint*.

Operational flexibility is the key to achieving this.

Well designed subsystem components and communications system will reduce complexity and increase flexibility.

The End