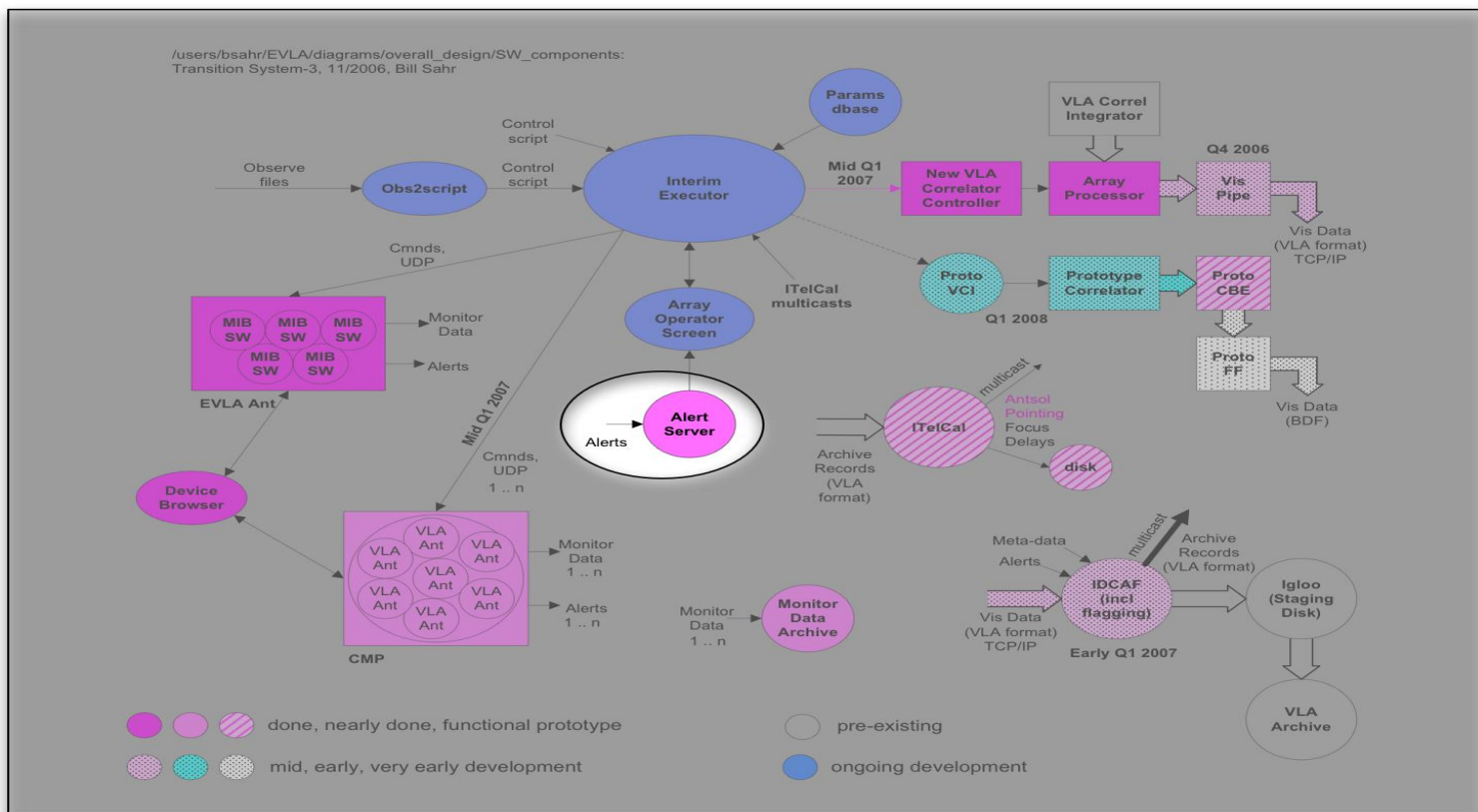
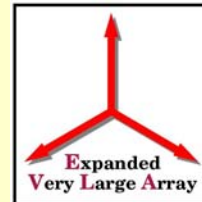


Alerts

Rich Moeser

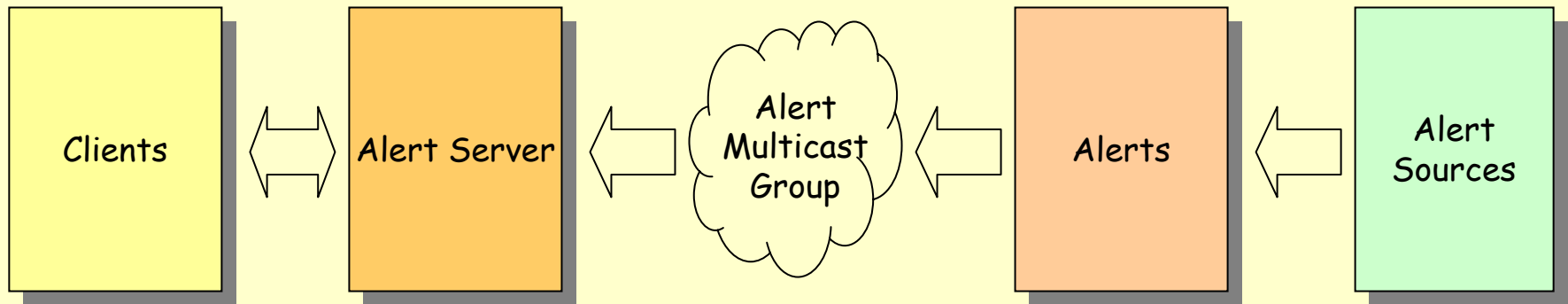
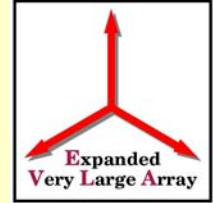


Place in the System



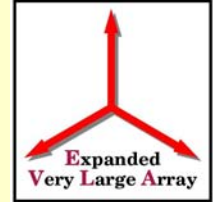


Alerts: Main Components and data flow





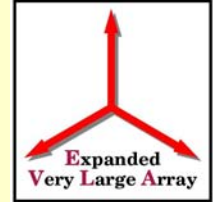
Alert Sources



- MIBs, CMIBs:
 - This is where most of the alerts originate
 - Located at
 - EVLA antennas
 - The AOC test rack (ATR)
 - Bench and desk tops
- CMP
 - Currently, only a subset of its monitor points are enabled.
 - The operators continue to rely on the ‘old’ Checker program for VLA alerts.
 - All of the required CMP alerts will be enabled when the MODCOMPs are turned off.
- Other Software Processes
 - Not many right now
 - mcwatch
 - It generates an alert when a test executor has been detected on mchost.
 - In the very near future, Telcal will be creating software generated alerts.



Alerts



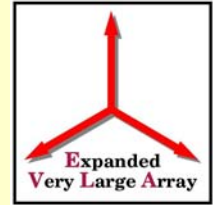
- What are alerts?
 - Alerts are XML messages sent by sources (MIBs, CMIBs, etc) as UDP multicasts
 - They indicate potential problems in the system that on some occasions require immediate attention
 - The XML alerts from MIBs conform to the XML schema file, evlamessage.xsd
 - The XML alerts from the Alert Server to client conform to the XML schema file, alerts.xsd (more on this later)

```
<xs:complexType name="monitorElement">  
  <xs:attribute name="name" type="xs:string" use="required"/>  
  <xs:attribute name="type" type="xs:string" use="optional"/>  
  <xs:attribute name="value" type="xs:string" use="optional"/>  
  <xs:attribute name="val" type="xs:string" use="optional"/>  
  <xs:attribute name="target" type="xs:string" use="optional"/>  
  <xs:attribute name="engr_unit" type="xs:string" use="optional"/>  
  <xs:attribute name="conv_type" type="xs:string" use="optional"/>  
  <xs:attribute name="slope" type="xs:string" use="optional"/>  
  <xs:attribute name="intercept" type="xs:string" use="optional"/>  
  <xs:attribute name="max" type="xs:string" use="optional"/>  
  <xs:attribute name="min" type="xs:string" use="optional"/>  
  <xs:attribute name="alert_arm" type="xs:string" use="optional"/>  
  <xs:attribute name="alert_on1" type="xs:string" use="optional"/>  
  <xs:attribute name="hi_alert_arm" type="xs:string" use="optional"/>  
  <xs:attribute name="lo_alert_arm" type="xs:string" use="optional"/>  
  <xs:attribute name="alert" type="xs:string" use="optional"/>  
  <xs:attribute name="hi_alert" type="xs:string" use="optional"/>  
  <xs:attribute name="lo_alert" type="xs:string" use="optional"/>  
  <xs:attribute name="a_period" type="xs:string" use="optional"/>  
  <xs:attribute name="s_period" type="xs:string" use="optional"/>  
  <xs:attribute name="o_period" type="xs:string" use="optional"/>  
  <xs:attribute name="aa_period" type="xs:string" use="optional"/>  
  <xs:attribute name="msg" type="xs:string" use="optional"/>  
</xs:complexType>
```



Alerts:

Sample alerts



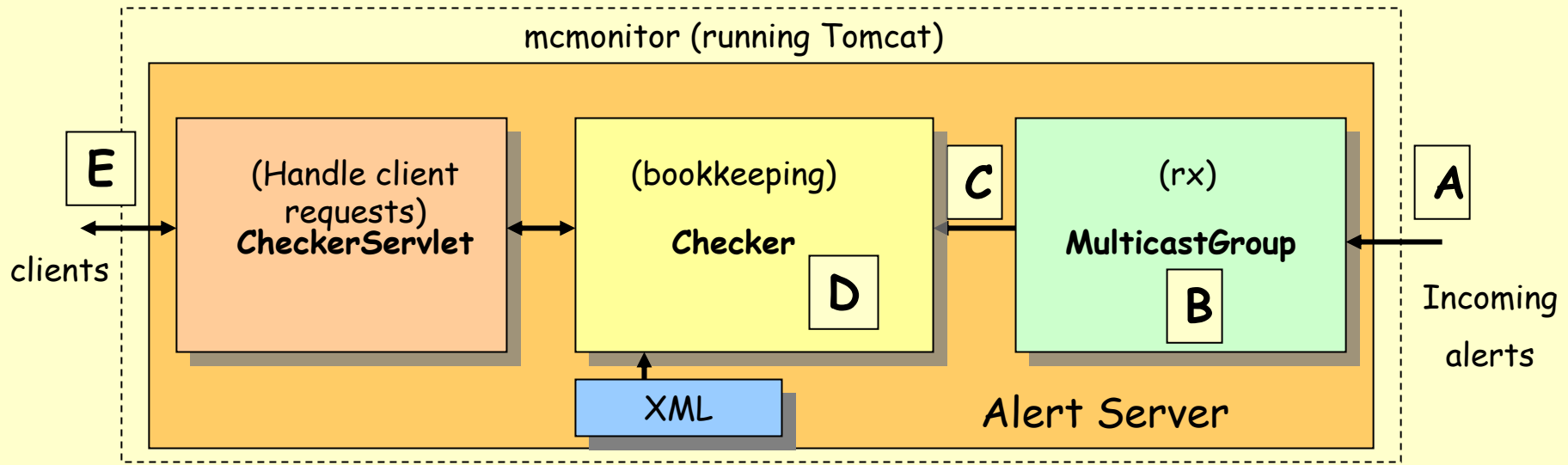
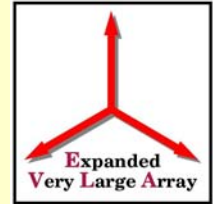
```
<EVLAMessage location='Antenna 13' timestamp='54067.8776686' >
<device name='T304' >
  <monitor name='VR_NEG7_T304' type='analog' value='-7.500560' alert='1' hi_alert='0'
  lo_alert='1' />
</device>
</EVLAMessage>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<EVLAMessage location='ea23-acu' timestamp='54066.7881133' >
<device name='ACU'>
  <monitor name='ELPosError' type='analog' value='-0.004087' alert='0' hi_alert='0' lo_alert='0' />
</device>
</EVLAMessage>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<EVLAMessage location='ea18-1302-1' timestamp='54066.7881827' >
<device name='L302' >
  <monitor name='M_LOCKH' type='digital' value='0' alert='0' />
</device>
</EVLAMessage>
```



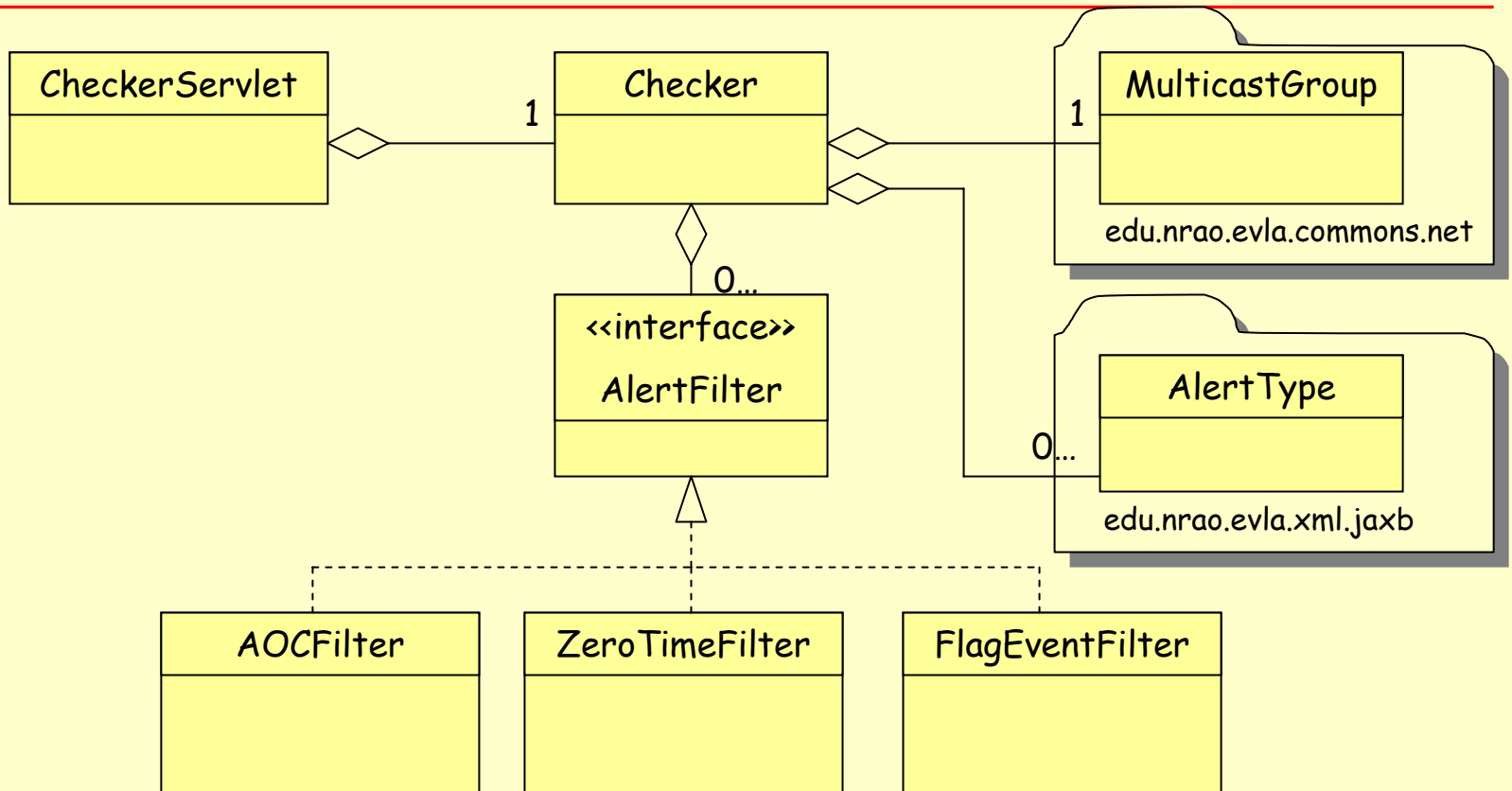
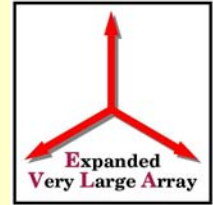
Alert Server



- A:** The multicast group receives an alert from the alert multicast group.
- B:** The multicast group uses JAXB to convert the XML message to a JAXB object.
- C:** The JAXB object is packaged into a MulticastGroupEvent and sent to Checker.
- D:** The JAXB object forwarded by the event is then converted to an Alert JAXB object and merged with its description and severity information obtained from the XML file. Then, depending on whether the alert is a new alert or a clear, Checker will add or remove the alert from the active alerts table.
- E:** CheckerServlet requests the list of alerts from Checker and sends the response back as XML. The conversion from JAXB object to XML is handled by Checker.

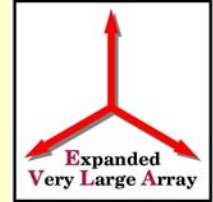


Alert Server: Class Diagram





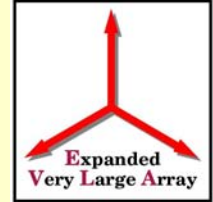
Alert Server: Alert Description File



- The information contained in the Alert Description File comes from Excel tech docs maintained by electronics.
- The document is formatted as XML.
- The alert attributes are...
 - devId
 - Device ID
 - mpId
 - Monitor point ID
 - severity (5 levels)
 - 0: Immediate: immediate maintenance is required at the equipment “Immediate Action Required”
 - 1: Standard: maintenance is necessary, antenna NOT usable, “Standard Antenna Rule”
 - 2: Minor: maintenance is necessary, antenna may be usable, “Write Maintenance Request”
 - 3: Archive Event: No Maintenance action necessary “Write an Informational Maintenance Report”
 - 4: Flagger Event: No Maintenance or operator action required
 - description
 - A description of the alert. Something the operators might understand.
 - action
 - A short message stating what – if any - action should be taken.
 - flag
 - true or false, if true, indicates the data is not valid when this monitor point is in alert



Alert Server: Alert Description File



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alerts>

  <alert antId="*" devId="ACU" mpId="AZCur2" severity="2" description="Azimuth Motor #2 High Current"
  action="This current should not occur in normal operation." value="" />
  <alert antId="*" devId="ACU" mpId="AZCur1" severity="2" description="Azimuth Motor #1 High Current"
  action="This current should not occur in normal operation." value="" />
  <alert antId="*" devId="ACU" mpId="ELCur2" severity="2" description="Elevation Motor #2 High Current"
  action="This current should not occur in normal operation." value="" />
  <alert antId="*" devId="ACU" mpId="ELCur1" severity="2" description="Elevation Motor #1 High Current"
  action="This current should not occur in normal operation." value="" />
  <alert antId="*" devId="ACU" mpId="AZVel" severity="2" description="Azimuth High Velocity" action="The
  antenna is moving too fast." value="" />
  <alert antId="*" devId="ACU" mpId="ELVel" severity="2" description="Elevation High Velocity" action="The
  antenna is moving too fast." value="" />

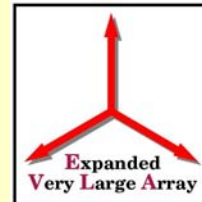
  <!-- FRM alert definitions (entered 10/24/05) -->
  <!-- FRM alert definitions (entered 06/29/06) -->

  <alert antId="*" devId="FRM" mpId="P5V" severity="2" description="+5V System Power Supply." action="Fault if
  off by ±300mV." value="" />
  <alert antId="*" devId="FRM" mpId="Gnd" severity="2" description="System GND." action="Fault if off by
  ±30mV." value="" />
  <alert antId="*" devId="FRM" mpId="P15V" severity="2" description="+15V System Power Supply."
  action="Fault if off by ±400mV." value="" />

  .....
```



Alert Server: alert.xsd Schema File



```
<?xml version="1.0"?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <xs:element name="alerts" type="AlertsType"/>

    <!-- Description of an 'alerts' element -->
    <xs:complexType name="AlertsType">
      <xs:sequence>
        <xs:element name="alert" type="AlertType" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>

    <!-- Description of a 'alert' element -->
    <xs:complexType name="AlertType">
      <xs:attribute name="id" type="xs:string" use="optional"/>
      <xs:attribute name="origin" type="xs:string" use="optional"/>
      <xs:attribute name="mjd" type="xs:string" use="optional"/>
      <xs:attribute name="antId" type="xs:string" use="required"/>
      <xs:attribute name="devId" type="xs:string" use="required"/>
      <xs:attribute name="mpId" type="xs:string" use="required"/>
      <xs:attribute name="severity" type="xs:string" use="required"/>
      <xs:attribute name="lo_alert" type="xs:string" use="optional"/>
      <xs:attribute name="hi_alert" type="xs:string" use="optional"/>
      <xs:attribute name="description" type="xs:string" use="optional"/>
      <xs:attribute name="action" type="xs:string" use="optional"/>
      <xs:attribute name="mpValue" type="xs:string" use="optional"/>
      <xs:attribute name="value" type="xs:string" use="required"/>
      <xs:attribute name="expire" type="xs:long" use="optional" default="86400"/>
      <xs:attribute name="flag" type="xs:boolean" use="optional" default="false"/>
    </xs:complexType>

  </xs:schema>
```



Alert Server: Filters



- Filters are implemented on the server-side to decrease the amount of data sent over the network.
- Filtering can be done on the client as well, but typically leads to duplicating source code and an increased packet size.
- Current Filters
 - aoc
 - Filters alerts that originate from the AOC.
 - zerotime
 - Filters alerts with a time of “00:00:00”. This is *usually* a result of MIBs sending alerts before time-synchronization
 - flag
 - Filters alerts that affect data quality. Monitor points that have “flag=true” in the Alert Description File.
- Others (do not exist yet)
 - dcs**
 - Filters alerts from the VLA antennas
 - time
 - Returns active alerts over given time range
 - severity
 - Returns active alerts for a specified severity



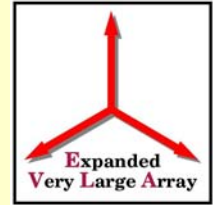
Alert Server: Communication



- The alert server uses HTTP for communication
- URI construction and parameters
 - <http://mcmonitor.evla.nrao.edu/evla-checker/?operation=<op>&filter=<filter>&id=<id>>
 - operation = “alertsinfo” or “alertclear”
 - filter = “aoc”, “zerotime”, or “flag”
 - id = the alert id (e.g., ea13.acu.azposerror) or “all”
- Examples
 - <http://mcmonitor.evla.nrao.edu/evla-checker/?operation=alertsinfo&filter=aoc>
 - <http://mcmonitor.evla.nrao.edu/evla-checker/?operation=alertsinfo>



Alert Clients



- Operator Alert Screen (more on this in a later talk)
- Flagger (does not use the alert server, listens to direct alert multicasts)
- Web Interface
 - <http://mcmonitor.evla.nrao.edu/evla-checker>