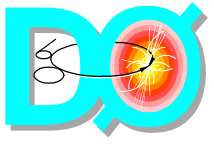


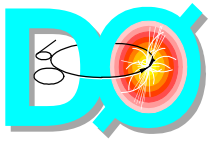
DØ Significant Event System

Geoff Savage
EPICS 2001



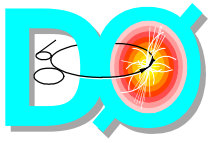
Outline

- **Overview**
- **Clients**
- **IOC implementation**
- **Message Contents**
- **Alarm Display Client**
- **Logger Client**



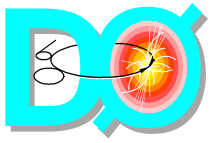
Overview

- Alarms are not the only events!
- Significant Event System
 - ◆ Monitor the current system state
 - ◆ Produce, distribute, display, and log events which are significant to the experiment
 - Alarms
 - Run state transitions
 - DAQ state transitions
 - ◆ Event types
 - Alarm
 - Information



Overview

- **Server**
 - ◆ **Written in Python**
 - ◆ **Contains the current alarm state of the system**
 - ◆ **Filters**
 - **Declared by receiver clients**
 - **Restrict transmitted messages**
 - **Filter dimensions**
 - ◆ **name, host, priority, severity, status ...**
 - ◆ **Current state available to receiver clients at connect time**



Overview

- **Clients**

- ◆ **API's in Python, C, and C++**

- ◆ **Sender clients**

- **Controls system (EPICS)**

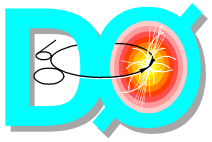
- **Host-level applications**

- ◆ **Receiver clients**

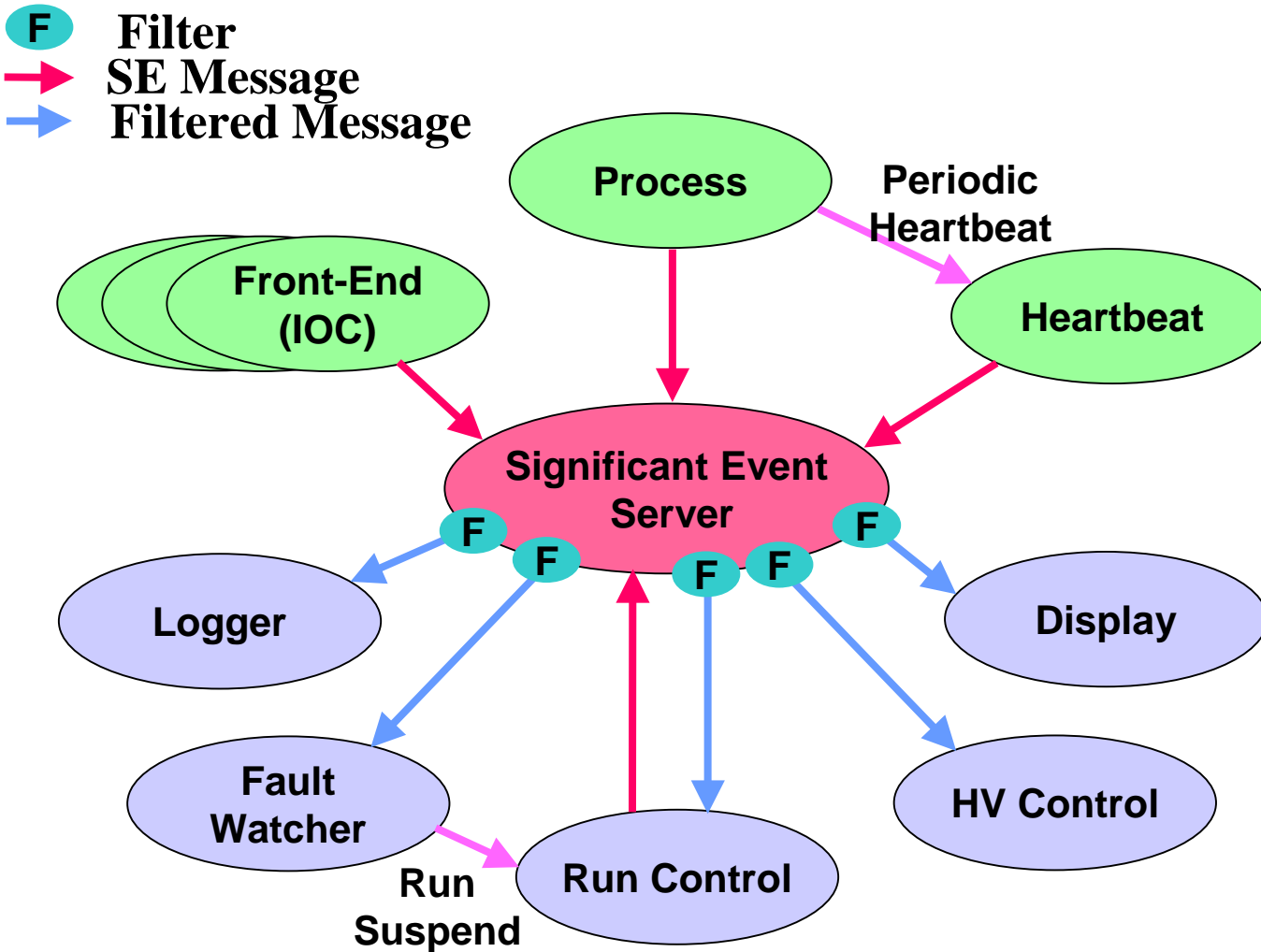
- **Multiple filters in server restrict messages sent**

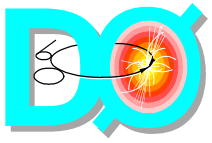
- **Filter dimensions**

- ◆ **name, host, priority, severity, status, ...**



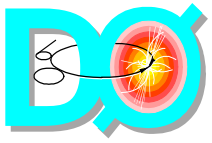
Overview





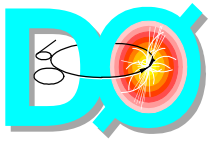
Clients

- **Sender clients**
 - ◆ **Control system (EPICS IOCs)**
 - Hook added to EPICS alarm processing
 - Single TCP/IP connection to the server
 - Event message sent when EPICS declares an alarm condition
 - ◆ **Host-level applications**
 - DAQ system components
 - Trigger system components



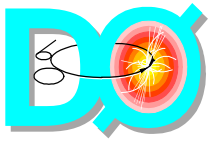
Clients

- Receiver clients
 - ◆ Operator alarm display
 - ◆ Global logger
 - Pass-all filter
 - ◆ Fault Monitor
 - Receives all alarms higher than a threshold priority
 - Suspends run when one or more alarm messages above the threshold are active (not acknowledged)



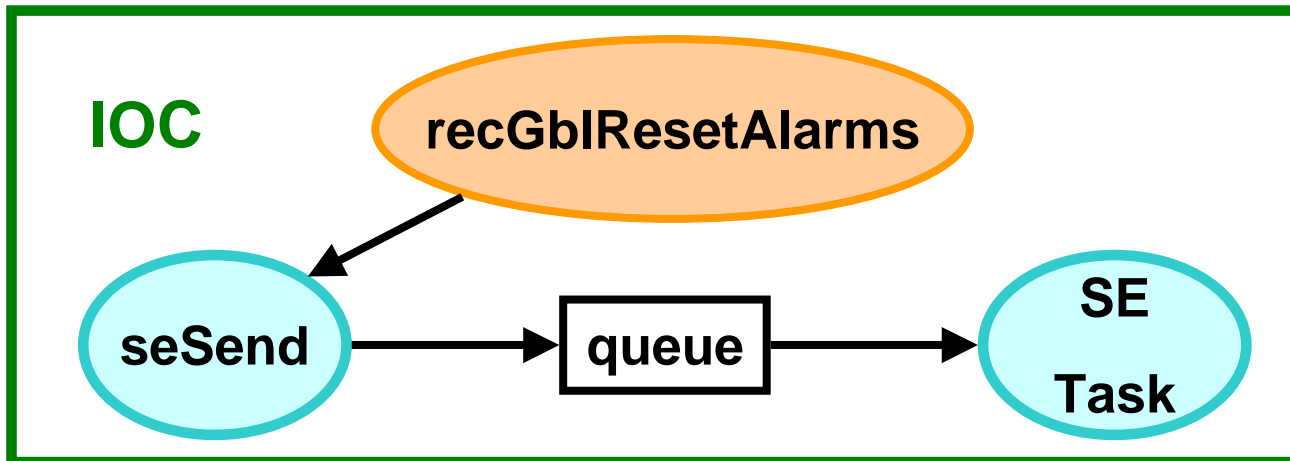
IOC

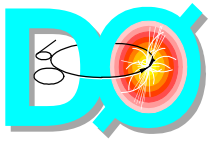
- **Send messages to server when alarms are determined by EPICS**
 - ◆ **Set alarm limits and conditions in records**
- **Ai, ao, longin, longout**
 - ◆ **return value, limits, status (source)**
- **Bi, bo, mbbi, mbbo**
 - ◆ **return state, status**
- **Generic (any others)**
 - ◆ **return status**



IOC - seSend

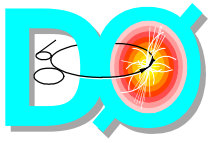
- seSend called by EPICS when an alarm is determined
- Connection to seTask through a POSIX queue
- Send PV name & volatile values





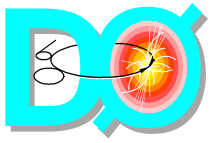
IOC - seTask

- **Maintain connection to SE Server**
 - ◆ **Initiate connection**
 - ◆ **Reestablish lost connections**
- **Create, format, and send alarm messages to server**
- **Add common EPICS fields**
 - ◆ **Message type (info, alarm)**
 - ◆ **Priority – importance in DAQ**
 - ◆ **Alarm message type**



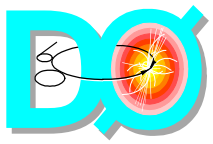
IOC - Configuration

- **In startup file**
 - ◆ **extHookInit** – connects seSend
 - ◆ **seStart** – connect to the server via TCP/IP link
 - ◆ **seWait** – wait for the SES to start before starting EPICS. Otherwise alarms may be detected before the server is active



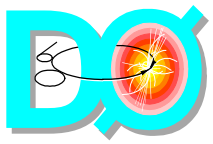
Message Contents

- Each SE message is a string with 13 space separated fields
 - ◆ Version
 - ◆ Timestamp
 - ◆ Message type
 - ◆ PV Name
 - ◆ Priority
 - ◆ IOC name
 - ◆ Database locator
 - ◆ Parent
 - ◆ Children
 - ◆ Transition
 - ◆ Severity
 - ◆ Alarm Type
 - ◆ Parameters



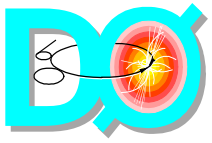
Message Fields

- **Version**
- **Timestamp**
 - ◆ **Seconds since the epoch**
- **Message type**
 - ◆ **Command, alarm, info, filter, l3**
- **Name**
 - ◆ **Must follow dzero convention**
- **Priority**
 - ◆ **0-255 – importance to experiment**



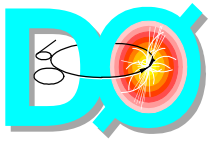
Message Fields

- **IOC name**
 - ◆ **Message source**
- **Database locator**
 - ◆ **Integer value**
 - ◆ **Only needed if name lookups are too slow**
- **Parent**
 - ◆ **For “smart” alarm acknowledgement**
- **Children**
 - ◆ **For “smart” alarm acknowledgement**



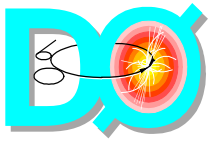
Message Fields

- **Transition**
 - ◆ **Bad (good -> bad)**
 - ◆ **Good (bad -> good)**
 - ◆ **Transient – notification that a problem was resolved**
- **Severity**
 - ◆ **Major, minor, invalid, no_alarm**



Message Fields

- **Alarm type**
 - ◆ **Binary, comment, analog Parameters**
 - ◆ **Based on alarm type**
 - ◆ **Binary**
 - **No extra data**
 - ◆ **Comment**
 - **Text string**
 - ◆ **Analog**
 - **Values or text**
 - ◆ **Different message class for each type**

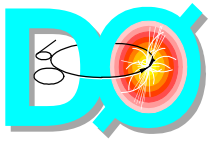


Alarm Display Client

The screenshot shows a software window titled "Alarm Display Client" with a menu bar containing "File", "View", "Settings", and "Help". Below the menu bar is a table with columns for "MINOR", "MAJOR", "INVALID", "ACK", and "GOOD". The rows represent different groups: "Control", "Online", "SDAQ", "CFT", "SMT", "CAL", and "MUO". The "Control" row has 22 Minor, 1 Major, 0 Invalid, 2 Ack, and 0 Good alarms. The "CFT" row has 2 Minor, 0 Major, 14 Invalid, 0 Ack, and 0 Good alarms. The "CAL" row has 122 Minor, 1 Major, 0 Invalid, 0 Ack, and 0 Good alarms. The "MUO" row has 232 Minor, 0 Major, 0 Invalid, 0 Ack, and 0 Good alarms. A "Status:" field is visible at the bottom left of the table area.

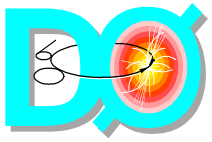
Group Name	MINOR	MAJOR	INVALID	ACK	GOOD
Control	22	1	0	2	0
Online	0	0	0	0	0
SDAQ	0	0	0	0	0
CFT	2	0	14	0	0
SMT	0	0	0	0	0
CAL	122	1	0	0	0
MUO	232	0	0	0	0

- Each button contains the number of alarms of a severity level that pass the filter for the row
- Alarms can appear in more than one button if they pass filters for multiple rows



Alarm Display Client

- **Description of columns**
 - ◆ **Bad alarms classified by severity**
 - **Minor**
 - **Major**
 - **Invalid**
 - ◆ **Ack – acknowledged alarms**
 - **Allows alarms to be ignored**
 - **Can unack**
 - ◆ **Good – after an alarm transitions from bad to good it remains in this column for five minutes then is removed**



Alarm Display Client

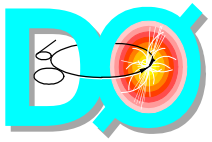
Group Name	MINOR	MAJOR	INVALID	ACK	GOOD
Control	22	1	0	2	0
Online	0	0	0	0	0
SDAQ	0	0	0	0	0
CFT	2	0	14	0	0
SMT	0	0	0	0	0
CAL	122	1	0	0	0
MUO	232	0	0	0	0

Status: |

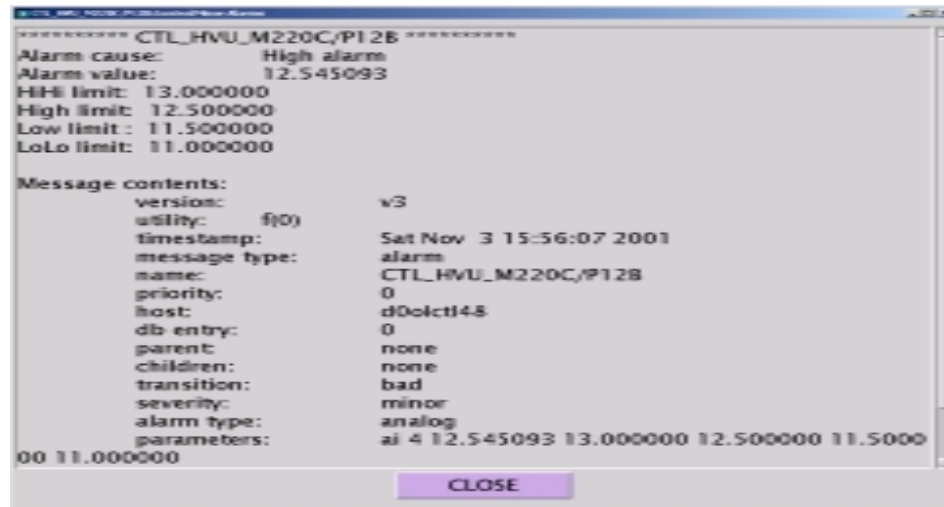
Left click a box to see the names of all the alarms in that category

Left click a name then left click the show button to see the message display

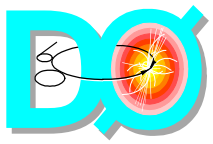
CTL_HVU_M220C/P12B	SHOW
CTL_HVU_M220C/M12B	ACK
CTL_HVU_M215B/M12B	ACK ALL
CTL_HVU_M215B/M12A	CLOSE
CTL_HVU_M217D/M12A	
CTL_HVU_M218C/P5D	
CTL_HVU_M221C/M12B	
CTL_HVU_M221D/P12B	
CTL_HVU_M221E/P12B	
CTL_HVU_M221E/M12A	
CTL_HVU_M221E/M12B	
CTL_HVU_M116B/M12B	
CTL_HVU_M215D/M12B	
CTL_HVC_07-00	
CTL_HVC_07-07	



Message Display



- Shows details on the alarm
- The contents of all the message fields are listed
- For alarms in the good column there can be multiple alarms with the same name



Logger Client

- Receiver client running as a daemon
- Receives all messages
- Writes messages in log files
 - ◆ Closes old file and opens new file at midnight
- Future version may log messages to Oracle database