



# Alarm Management 101

Jarid Kling

Alyeska Pipeline Service  
Company

# Objective

- Share our discoveries in understanding the need and issues with implementing alarm management
- Identify critical steps in initiating, implementing, and maintaining an Alarm Management Program
- Share some of the lessons learned in our journey

# Why

- Most companies understand the need for alarm management (?):
  - Lowers risk of Controller Error
  - Places accountabilities where they belong
  - Documents and Justifies Companies Operational Philosophy
  - Improves overall reliability, operability, and maintainability of pipeline
  - Required by Law

# Driver for Alarm Management at Alyeska

- Major equipment modifications in field
  - Essentially new pump stations
  - Move from manned to unmanned stations
- Volume/Frequency/Unfamiliarity of Alarms
- Paradigm shift in operations, maintenance, and engineering
- Legacy Issues

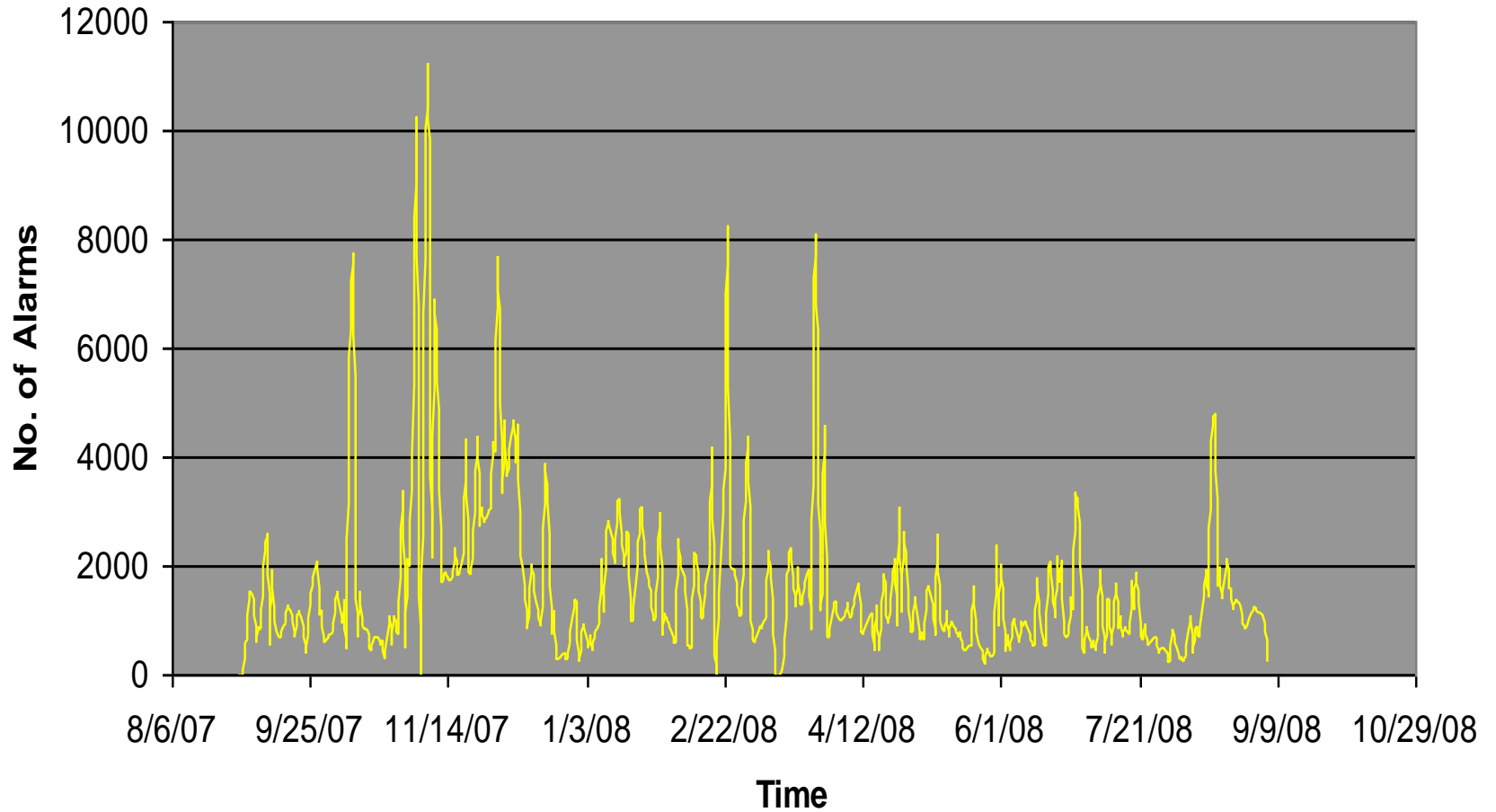
# Legacy Issues

- With over 30 years of ongoing projects, maintenance, and other modifications along with a lack of tight controls over what information goes to a Console, we found ourselves at an unacceptable risk level.
- Resulted in Controllers only having the ability/time to respond to the highest level and some medium level alarms due to the alarm rate. Getting to this point was a slow chronic process that took over 30 years.

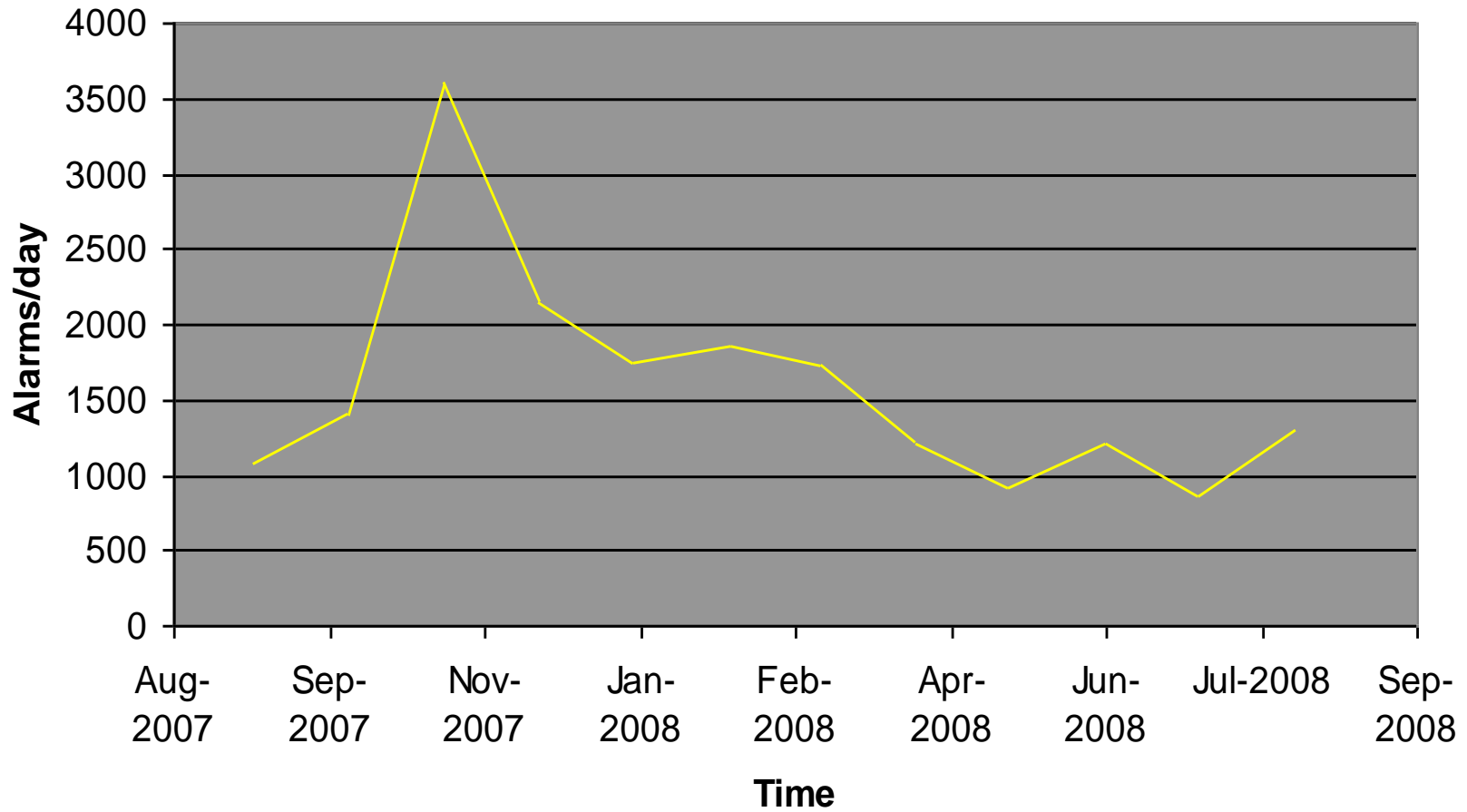
# Breaking Point

- Recent project work added roughly 7000 alarm points at each new station in order to unman them.
- One Mainline pumping unit has ~500 I/O
- Project philosophy was that all information would go to the Control Center – OCC could figure out what to do with it.
- At one point we had over 15,000 alarms in one day

# Alarms Per Day Aug 2007 - Sept 2008



# Monthly Averages Aug 2007-Aug 2008



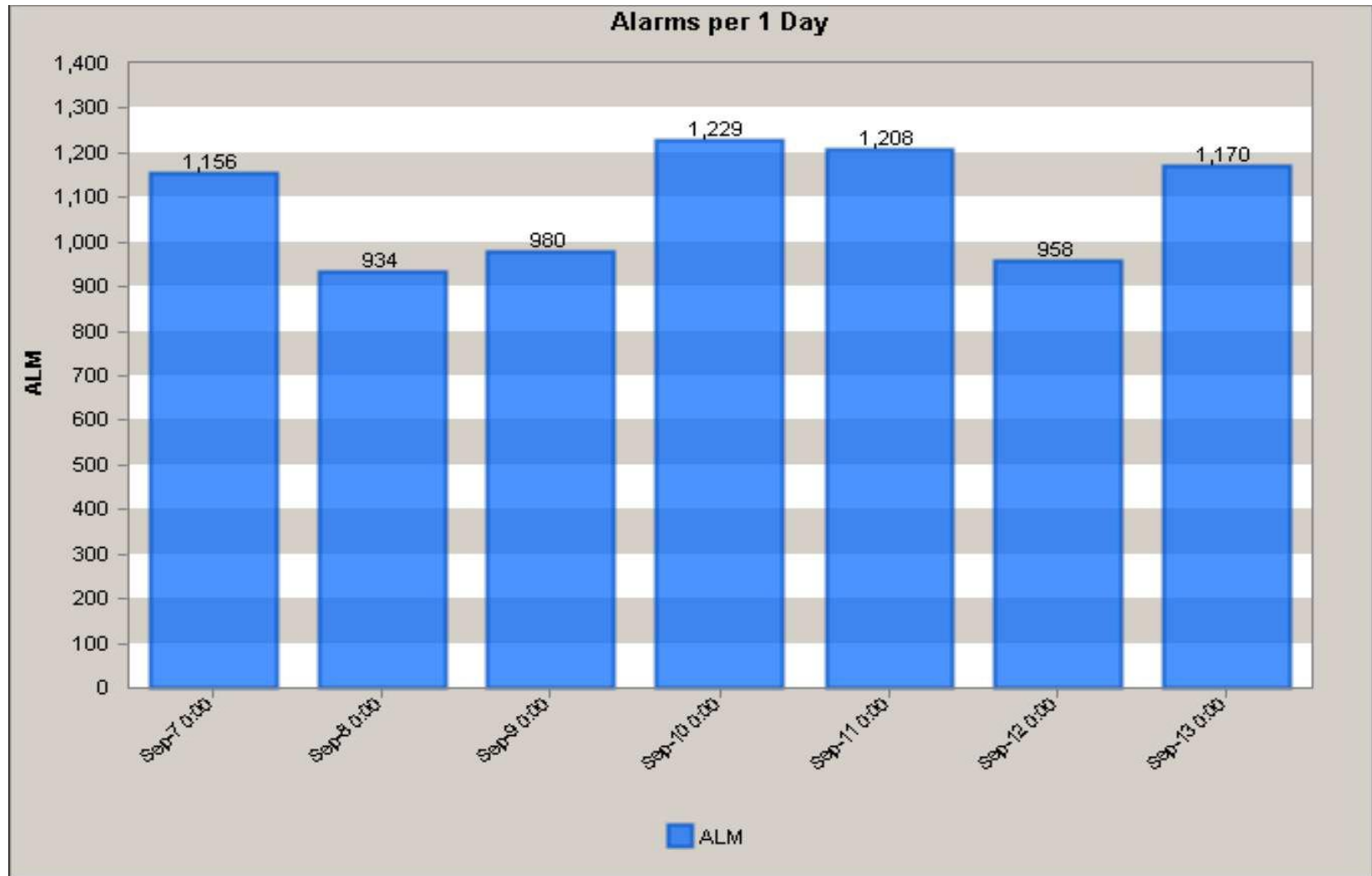
# Where Does it Start

- Controllers were primary force behind conveying the problems to management – as it should be.
- Literature research into Industry Best Practices, standards, etc. Accelerate the learning curve.
- Qualitative and Quantitative Assessment must be done (risk assessment?) on each console – How bad is the problem?
- Process must have management/executive buy-in and support from the start!
- Develop project team (3-4 primary members)
- Benchmarking with other industries or pipelines
- Process must be initially driven by Controllers

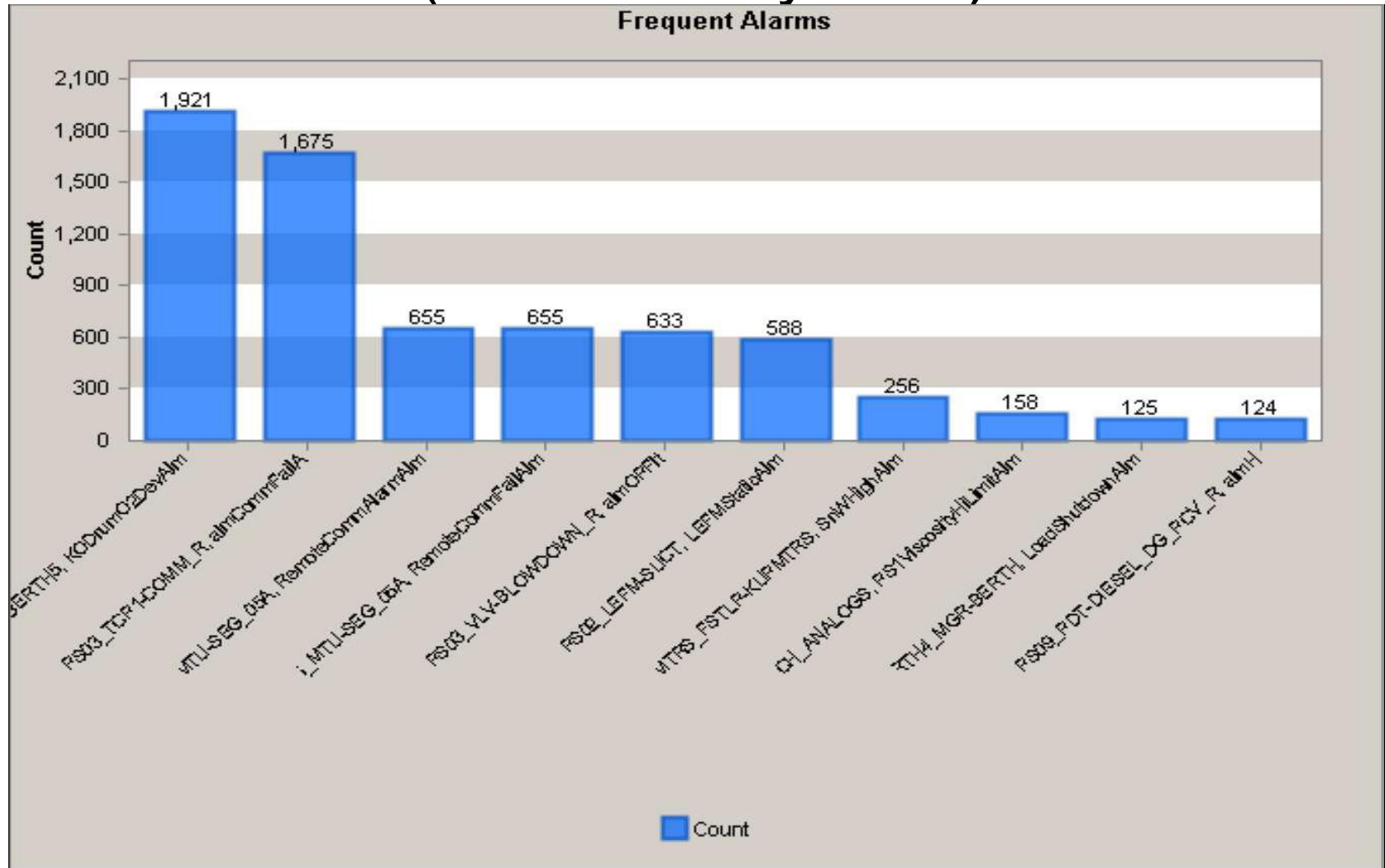
# Alarm Management is a Process – Not an Event

- The process will take time (years) and it will require resources (varies depending on stage)
- Critical to the process is having the tools to measure system performance
  - Data must be readily available and easy to access
  - Weekly reports to senior management are essential

# Alarms Per Day



# Weekly Frequent Alarms (for entire system)



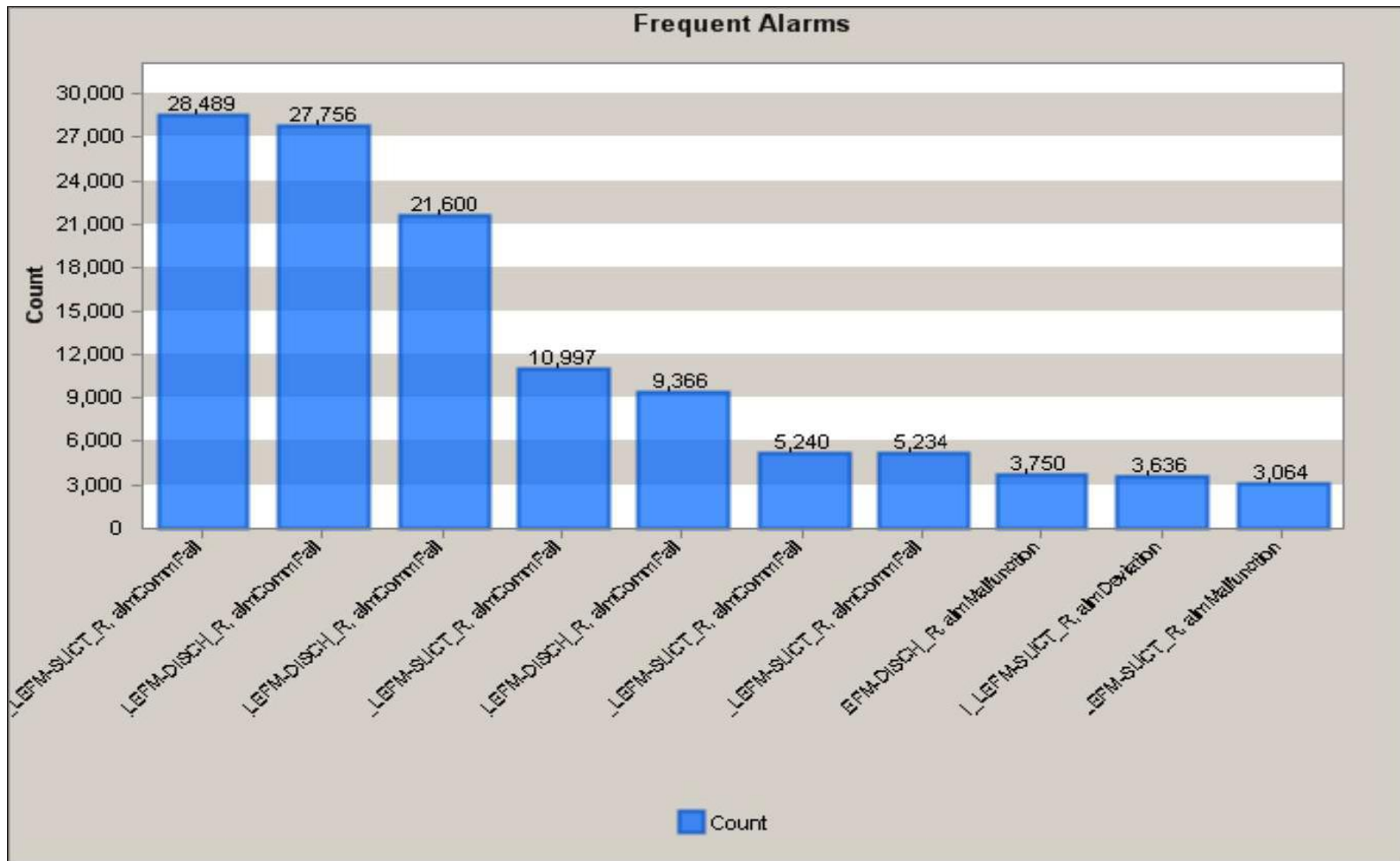
# Weekly Data

Event	Alarm	Count	Percent	Cumulative %	Priority
COMPRSR_ALM-BERTH5	KODrumO2DevAlm	1921	15.61	15.61	5
PS03_TCP1-COMM_R	almCommFailA	1675	13.61	29.23	2
PS05_MTU-SEG_05A	RemoteCommAlarmAlm	655	5.32	34.55	3
PS05_MTU-SEG_05A	RemoteCommFailAlm	655	5.32	39.87	2
PS03_VLV-BLOWDOWN_R	almOPFlt	633	5.14	45.02	2
PS02_LEFM-SUCT	LEFMStaticAlm	588	4.78	49.80	2
KUPMTRS_FSTLP-KUPMTRS	SnWHighAlm	256	2.08	51.88	3
PS01_TVP-DISCH_ANALOGS	PS1ViscosityHiLimitAlm	158	1.28	53.16	3
BERTH4_MGR-BERTH	LoadShutdownAlm	125	1.02	54.18	5

# Other Data

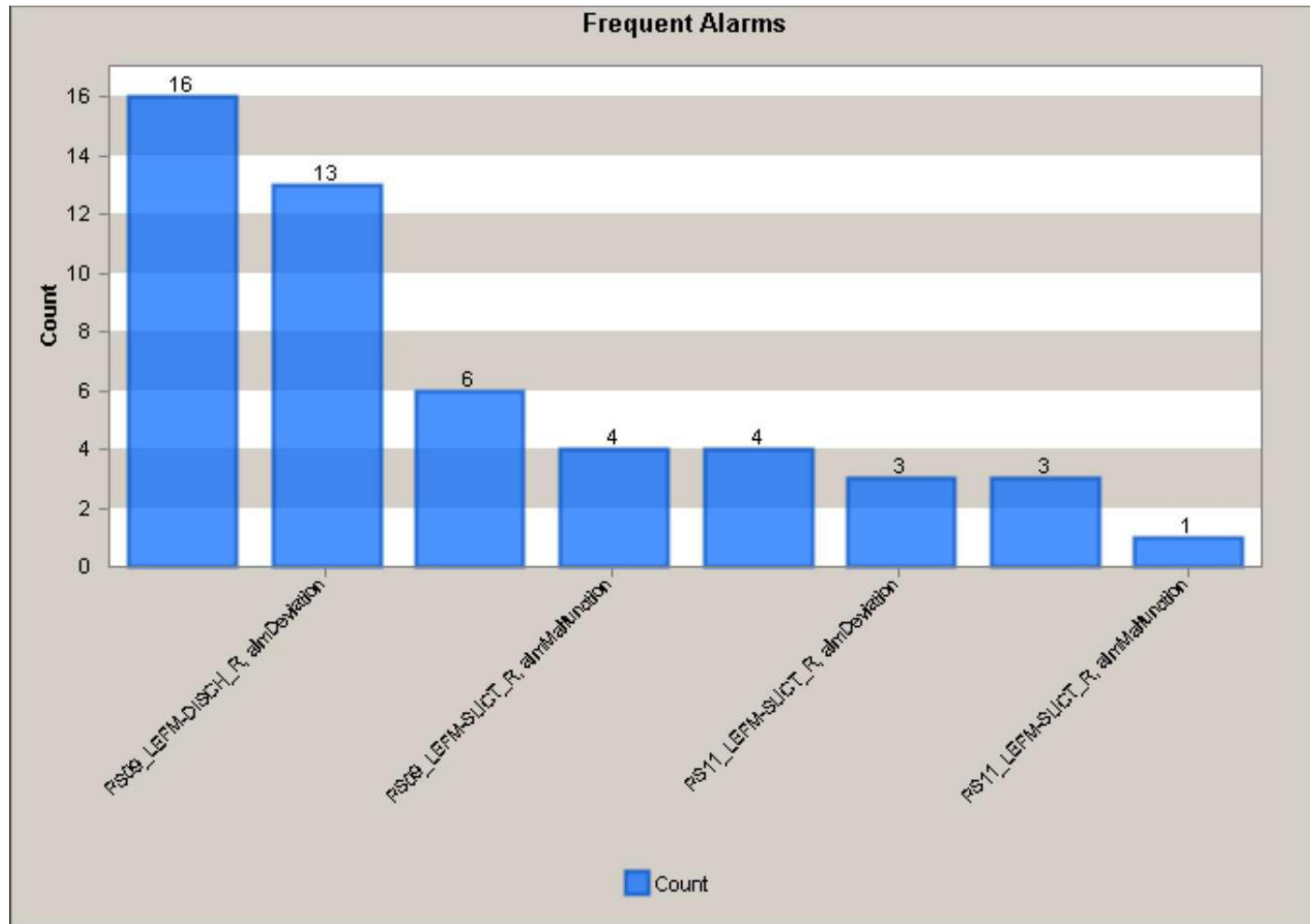
- Number of alarms/day
- Frequent alarms
- Consequential alarms
- Flood Events
- Alarm data on specific systems (leak detection, pump station, power generation, etc.)

# Year to Date for LEFMs





# Most Recent Weekly Report



# Alarm Analysis Software is Critical to Alarm Management

# Corporate Standards

- A Control Room needs a clearly defined and documented Alarm and HMI Standards based on industry best practices
- These standards need to be understood and followed by those groups which can impact your control systems and HMIs (engineering, projects, maintenance, etc.)
- Without a standard, these groups are left to using their best professional judgment on what information does/does not to go to a Controller.

# Items to Consider in a Standard

- Alarm philosophy
- Definitions and criteria
- Documentation/Rationalization process
- Annunciation and Response
- Alarm Handling
- Alarm Design Considerations
- MOC
- Training
- Alarm Maintenance Process
- Performance Criteria

# Alarm/Event Database

- Database is needed to document all pertinent information about each of your alarms/events (~100-150 alarms/day)
  - Description of alarm
  - Purpose of alarm (justification)
  - Typical response
  - Consequence of alarm going unacknowledged
  - Time to respond
  - Priority of alarm/event
- This information allows you to determine who is the appropriate group to respond - four basic groups:
  - No response needed (log only)
  - Engineering response
  - Maintenance response
  - Controller response
- The primary goal is to ensure that the right information is going to the right people so they can act on it!

# System Integrity

Control Room → Operability

Maintenance → Maintainability

Engineering → Reliability

# Milestones

- The first major reduction in alarms should come from the removal of events that are not pertinent to Controllers.
- The next stage is addressing nuisance and frequent alarms – continuous process
- State-based suppression is one of the final stages which requires major involvement from engineering and programming

# Alarm Maintenance

- Ongoing management needs to be driven by Automation/SCADA with input from the Control Room.
- Weekly meetings with Engineering, SCADA, and Controllers.
  - Reviewing Bad Actors
  - Assigning actions
  - Resolution
  - Suppression issues

# Fallout

A second paradigm shift for the company...

- Instead of controllers spending most of their time responding to alarm floods, they can focus on the operation and be better prepared to respond to AOCs, look for improvements, efficiencies, etc.
- With information going to the right people, they are taking that information and evolving the business
- Moving from an alarm database to an information database
- Converging events with historical data to drive improvements

# Lessons Learned

- Match the personality types to the tasks – right skills for the job (tactical vs strategic, big picture vs details, etc.)
- Realization of it being a program vs project
- Get the right tools to measure performance
- Assign and ensure that the resources (people, time, tools, and money) are available
- Willingness to compromise
- Have patience – this process takes time, but it will reap many rewards

# Summary

- Learn as much as possible from literature and benchmarking
- Get Management support
- Create your standards and apply them
- Get analysis tools and use them
- Apply the right resourcess
- Do the analysis and create/modify your database
- Implement weekly reviews of the system, assign action items, and **hold people accountable**

Q&A