What an Alarm System should do

An Alarm system is needed to notify any important incidence in the control system, but also to keep a continuous feedback of machine and beamlines operation. Logging and notification of changes in operation conditions may prevent problems before they occur.

Our requirements were focused on having a reliable notification service, capable of logging and absolutely integrated in our control system. Although two alarm logging systems already existed in Tango as independent tools, at ALBA we preferred to integrate alarms in the existing applications. To develop our own alarm system we borrowed as much as possible from the existing systems.

The Tango Alarm System

The Tango Alarm System [5] was developed at Elettra institute by Graziano Scalamera and Lorenzo Pivetta. It uses a MySQL alarm database containing sets of rules that are permanently checked by a central daemon, the Tango Alarm Server. This server logs alarm changes and triggers actions if needed. The rules are combinations of boolean operators and Tango Attribute values.

Soleil Alarm Database

Soleil has developed an Alarm Database that mimics the behaviour of Tango Archiving System [6], but focused on storing Attribute Quality (Valid/Invalid/Changing/Warning/Alarm) changes instead of values. The system uses a MySQL database and a pool of Archiver devices polling the quality of those Tango Attributes registered in the Database (events are not supported).

Alarm's Declaration with Panic

PyAlarm applies python parsing; enabling a richer rule syntax with list comprehensions, regular expressions, string replacement and other functional features.

Source of an Alarm

PyAlarm applies python parsing; enabling a richer rule syntax with list comprehensions, regular expressions, string replacement and other functional features.

The source of an alarm comes from:

- Vacuum
- Temperatures
- EPS devices
- Availability of services
- Any Tango attribute...

Types of messages

- **Alarm**: The alarm condition has been activated.
- **Recovered**: Alarm conditions are now inactive, but alarm state is kept.
- **Reminder**: When Condition is still active for X period or it is active again after a Recovered period.
- **Acknowledged**: Alarm has been reset by operator.
- **Auto-reset**: Alarm has been reset after being in Recovered state for a long time.

Alarm Life Cycle

In ALBA Alarm system every Alarm requires human acknowledge to be reset. We added additional alarm notifications to avoid alarms being unnoticed. Changes in alarm condition value will trigger Reminder or Recovered emails while it is not acknowledged yet.

Alarms Log and History

- **ALARM**
- **WARNING**
- **DEBUG**
- **ERROR**

Alarm Device Properties

Poster Contacts: Sergi Rubio (srubio@cells.es), Fulvio Becheri (fbecheri@cells.es), David Fernández (dfernandez@cells.es)