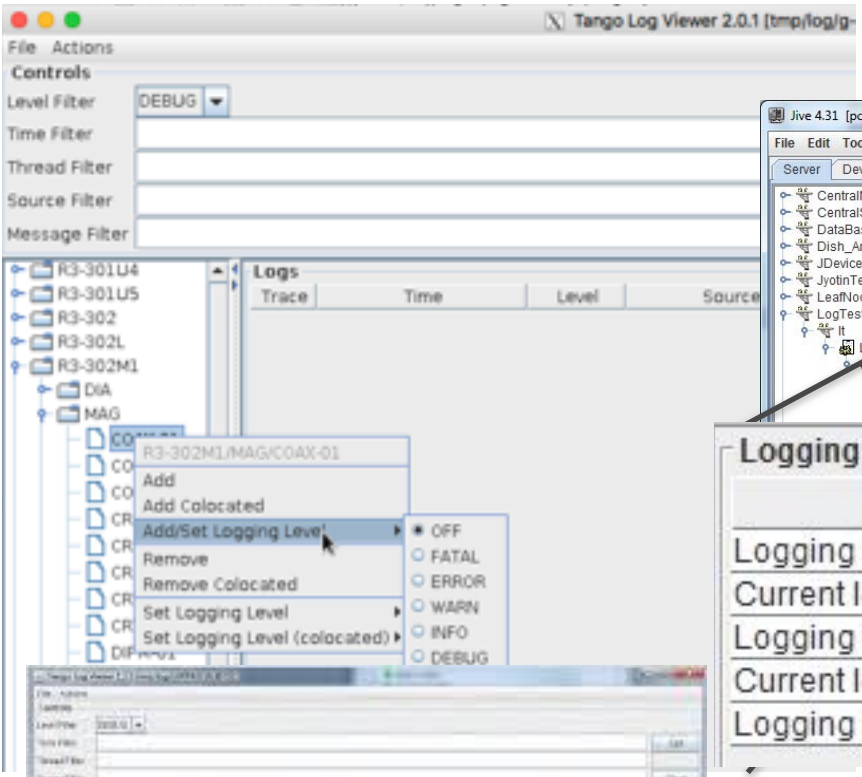

MAX IV

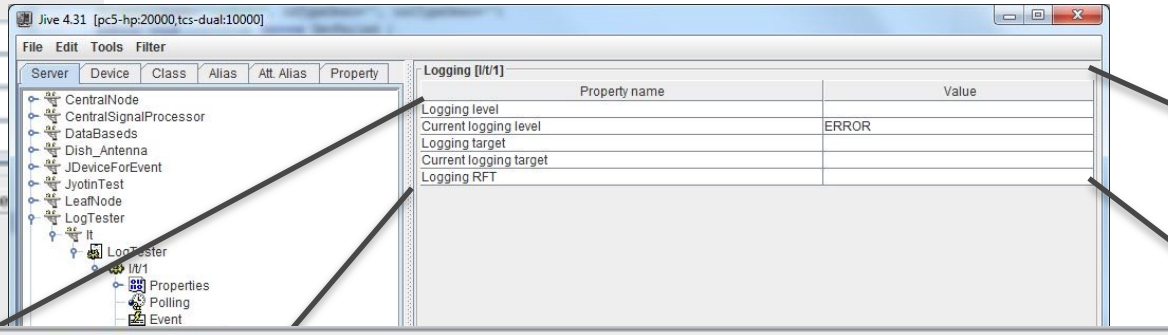
Monitoring and Alarms in MAX IV

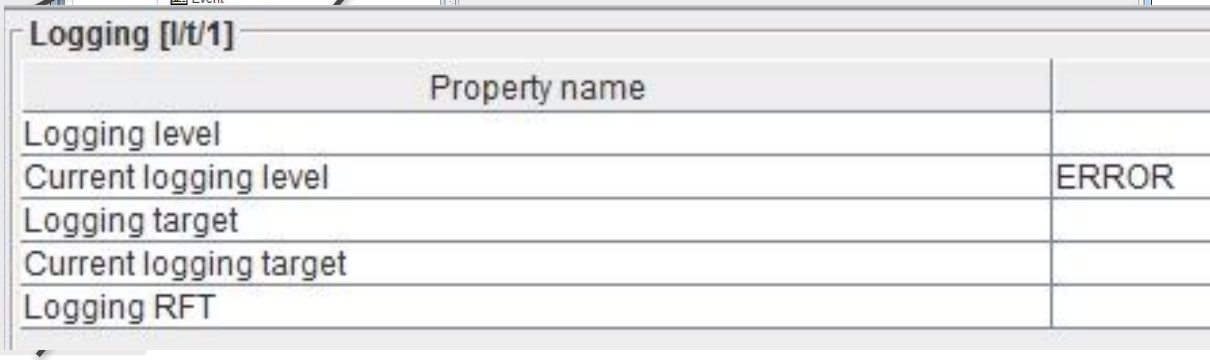
… on behalf of KITS

Tango Workshop ICALEPCS 2017, Barcelona

Tango Logging

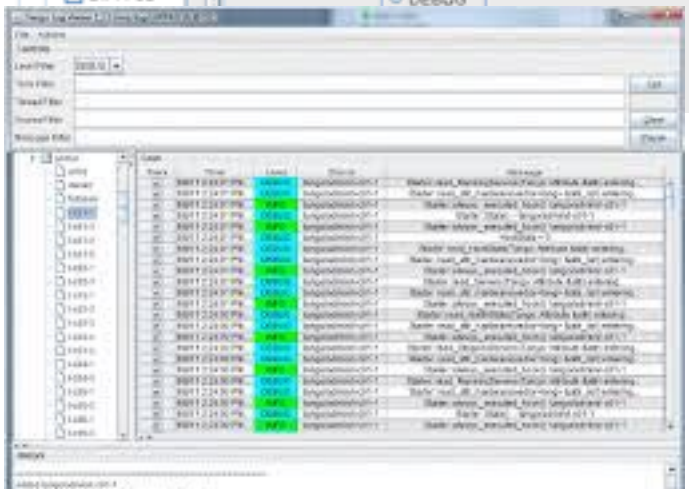






Logging [l/t/1]

Property name	
Logging level	
Current logging level	ERROR
Logging target	
Current logging target	
Logging RFT	



Topics

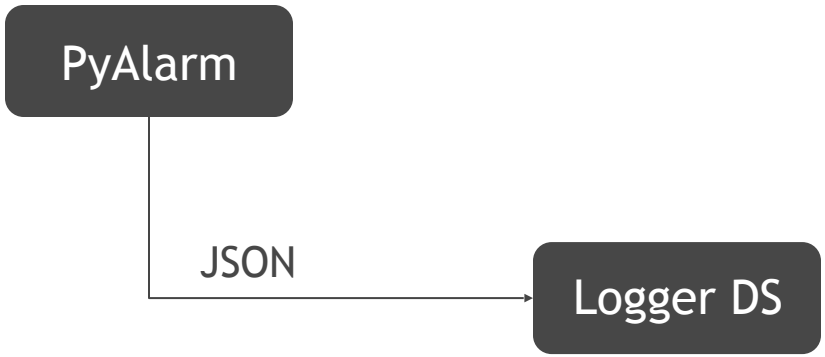
- Alarm System
- Monitoring System

Alarms Overview

- Panic
- Logger DS
- Elasticsearch
- Kibana 3

Panic

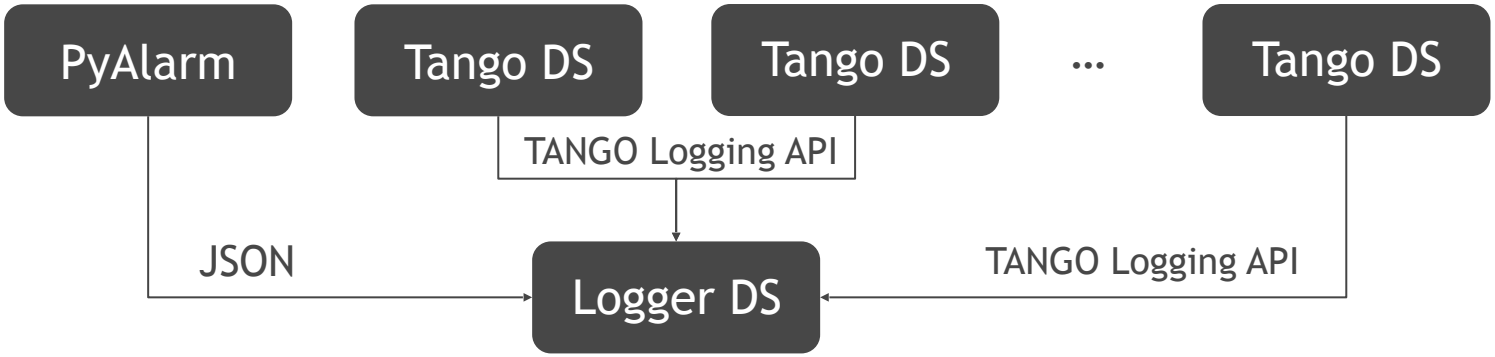
- PyAlarm: Alarm tango device server.
- New property added: LoggerDevice. It points to the tango device server to act as a logger.
- When an alarm is generated, a JSON message will be sent to the logger device server with all the relevant information.



Available in Github: https://github.com/tango-controls/PANIC

Logger DS

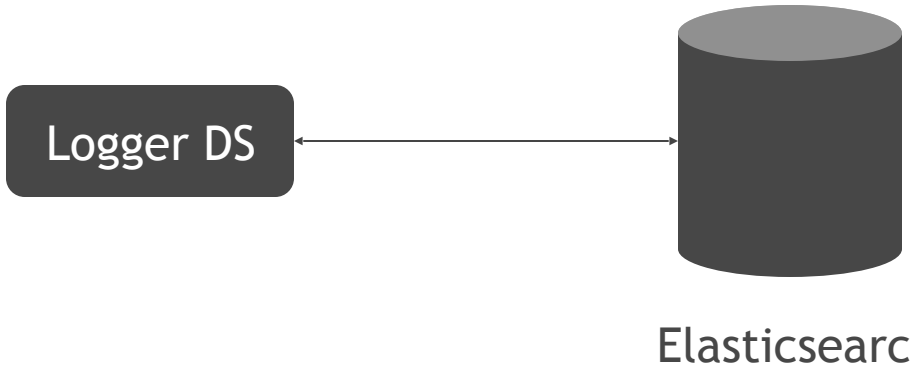
- Tango Device Server written in Python to handle logs.
- Alarm command: takes a JSON string and send it to Elastichsearch.
- Log command: compliant with the TANGO logging API. It inserts general TANGO logs into Elasticsearch.
- It can be used as a logging target for other Tango device servers.



Available in Github: https://github.com/MaxIV-KitsControls/dev-maxiv-logger

Elasticsearch

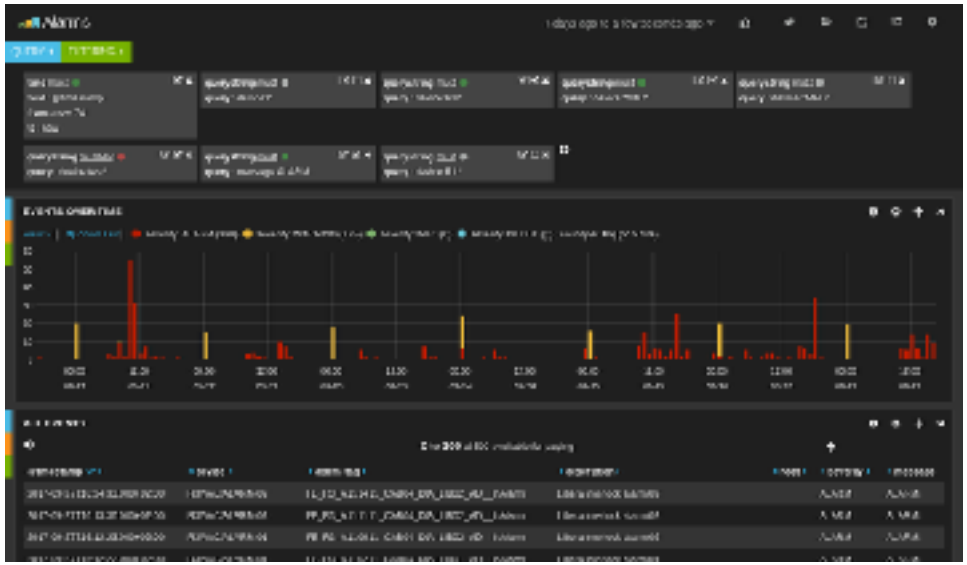
- Distributed, RESTful search and analytics engine.
- Alarms stored in Elastichsearch for long term support.
- Scalable to multiple hosts if needed.
- The logger device generates a new index in for each day.



More info here: https://www.elastic.co/products/elasticsearch

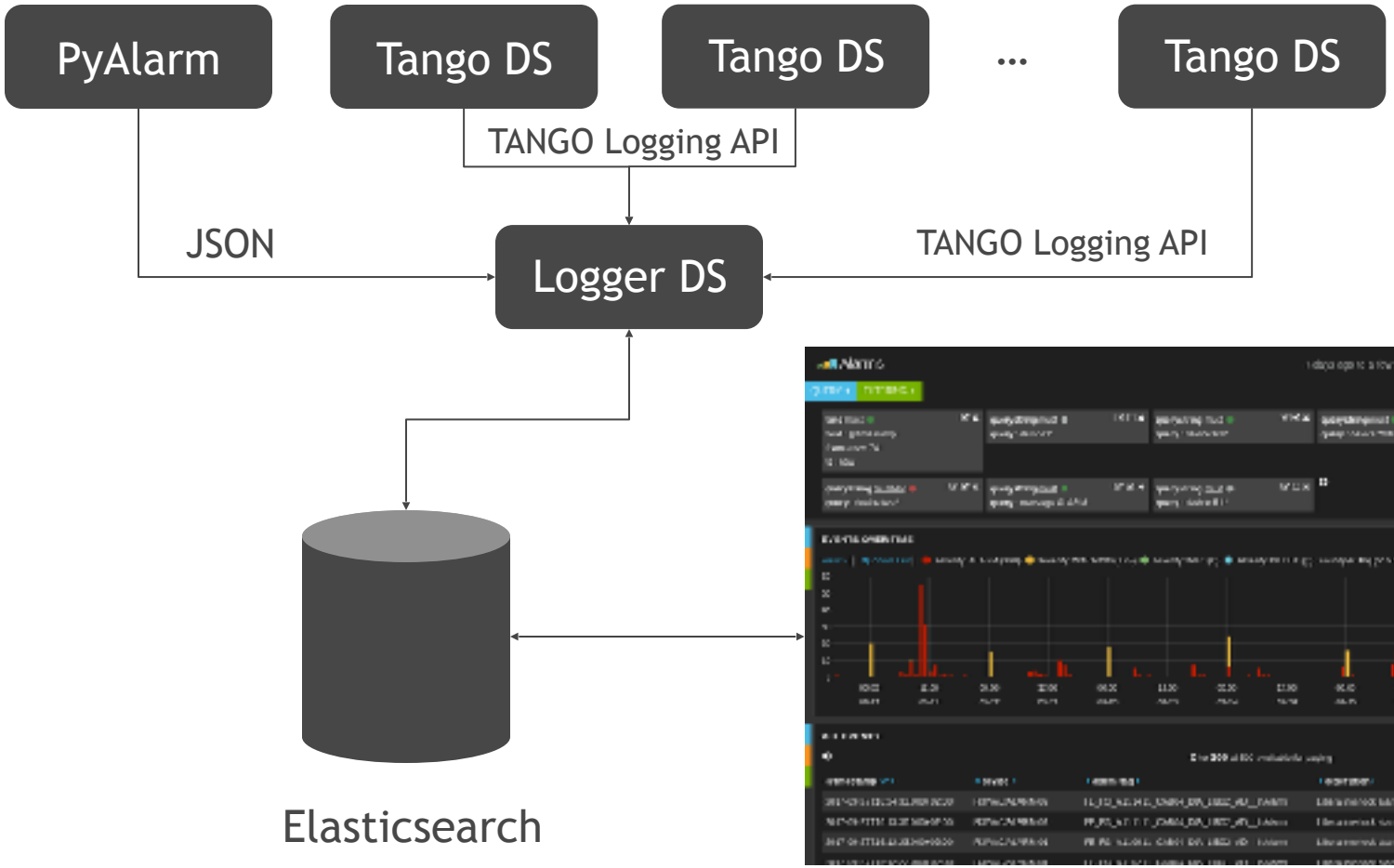
Kibana 3

- General web interface to access to Elasticsearch data.
- Configured to show the output of any tango device using the logger device as a logging mechanism.



More info here: https://www.elastic.co/products/kibana

Software Architecture



Monitoring Overview

- Prometheus
- Node Exporter
- Tango Exporter
- Libera Exporter
- Interfaces
- Alarm Manager

Prometheus

- General server monitoring system.
- Easy to set up, thanks to not needing any external database and to its "pull" model which means that there is no configuration needed on the monitored machines, only on the central server.
- It supports alerting through emails etc, via the alertmanager sister service.

More info: https://prometheus.io/

Node Exporter

- The general monitoring on each host is done by a service called "node_exporter".
- It needs to be installed on every host that is going to be monitored.
- It requires no configuration, although there are different plugins available.

Tango Exporters

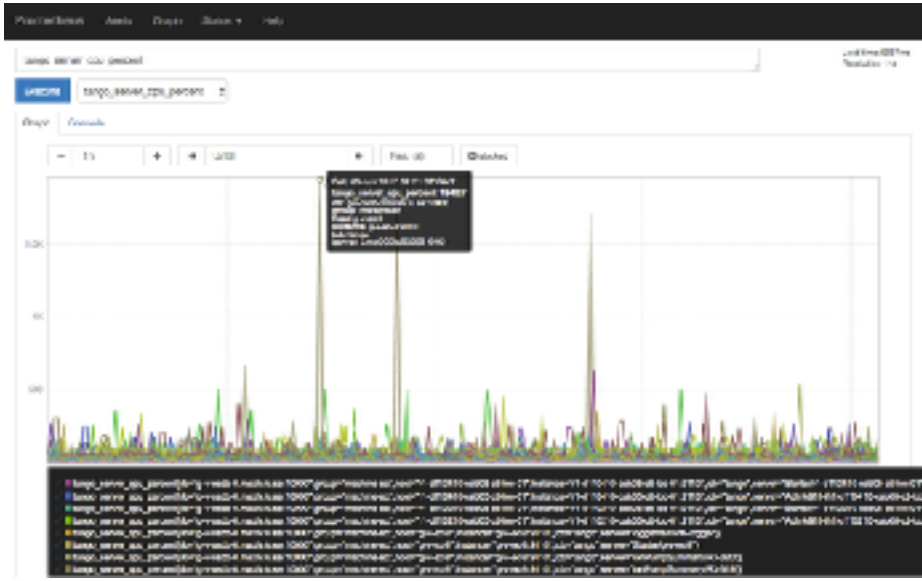
- tango_exporter: service written in python to check the starter device for servers that should be running.
- It monitors those servers for some process specific metrics (cpu usage, memory, etc.).
- This service needs to be running on all servers running tango devices.
- It has no configuration.

Libera Exporter

- Libera-exporter was developed in order to move platform monitoring from Libera Tango devices to prometheus.
- It supports up to 200 metrics like (module temperature, voltage, current etc..)

Interfaces

Grafana custom dashboard



Prometheus built in interface

More Grafana info: https://grafana.com/

Alert Manager

- Prometheus can be configured to continually detect various conditions and raise alerts if they evaluate as true.
- Alerts are specified in a config file, using the prometheus query language.
- Normally the alerts are just evaluated and stored as individual metrics in prometheus.
- The **alertmanager** is a separate service that Prometheus can talk to, and which takes care of sending out messages (e.g. emails) about alerts.
- It can also filter and group the messages to limit the number of messages sent out, etc.

Monitoring Architecture