EPICS Alarm System in Phoebus

Peter Bonneau
2022-04

EPICS Alarm System in Phoebus

I am developing an EPICS alarm system based on CS-Studio Phoebus. Phoebus will be used for new EPICS system development and will replace the existing Eclipse-based CS-Studio systems as detailed in my note DSG Note 2021-37 and talk DSG Talk 2021-17.

I am working on implementing a message streaming interface between the Phoebus user interface (UI) and the alarm server. This message streaming interface establishes a communication link used for entering the alarm server configuration settings via an UI menu for each process variable (PV) monitored by the alarm system. This link is called the *configuration stream* and is shown in the block diagram on slide #2.

To create the configuration stream, I wrote a script based on a template found in the Phoebus source code. The *create_alarm_topic.sh* script defines the operational parameters of the configuration stream. The parameters include the name of the stream, message size, and the location of the log file. In Kafka jargon, a topic is the unique name of a specific message stream.

The configuration stream uses the Apache Kafka infrastructure that I have developed, configured, and initialized (DSG software memo 2022-03). The script is used only once at the creation of the alarm system to initially program the *Kafka server* (FIG. 1) with the configuration stream parameters. I executed the script and no errors were reported by the *Kafka server* or the Kafka *cluster management*.

To test the configuration stream, I opened a PV alarm configuration menu in the Phoebus UI (FIG. 2) and entered alarm system settings for a test PV. The settings included guidance on how to respond to an alarm and links to user interface displays. I saved the settings and closed the UI menu. When UI settings are saved, the configuration stream transmits the entries from the UI menu and writes it to the *alarm server* (FIG. 1).

- **Developing CS-Studio Phoebus based controls, monitoring, and alarm system - to be implemented on Hall C detectors**

- **Debugging message streaming between the Phoebus user interface and the alarm server (block diagram on slide #2)**

- **Plan to implement the alarm configuration logger as the next development of the alarm system**

EPICS: CSS-Phoebus

To verify that the PV settings were written correctly, I used the *alarm server* command console to read-back the alarm settings for the test PV. The name of the PV and several other settings were read-back correctly, however most of the settings were not saved. I checked the *Kafka server* console and the log files for errors and none were listed.

The streaming messages are text-based however, Phoebus does not have a program that will directly display the text of a message. To help in the debugging of this issue, I wrote a program using Kafka script commands to continuously monitor the configuration message stream. Since the configuration messages are text, being able to read them directly could help in identifying the source of the error.

I discovered via my Kafka message monitoring program that the Phoebus alarm UI was sending a message, however the *Kafka server* was not accepting the message. I researched the configuration of the Kafka server and found six of the parameters in the *create_alarm_topic.sh* script were not in effect. In further research, I found that the script template in the Phoebus source code was out of date with the Kafka 2.13-3.1.0 version that was recommended in the Phoebus documentation.

After correcting the script to match Kafka 2.13-3.1.0 and rerunning, the configuration message stream worked correctly. The menu shown in FIG.2. shows a PV configured to be monitored by the alarm system.

I also wrote a *alarm_preferences.properties* file that customized the UI menus for Hall C NPS as shown in FIG.3. Phoebus was recompiled to implement the custom menus.

I plan to implement the *alarm configuration logger* (FIG.1) as the next development of the alarm system.
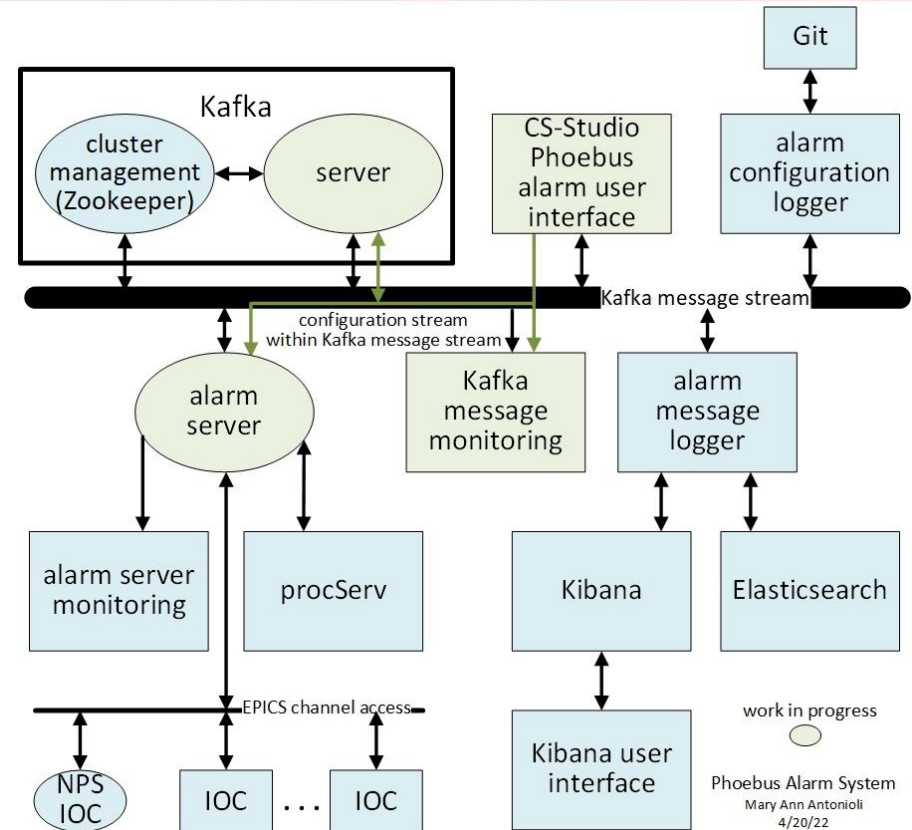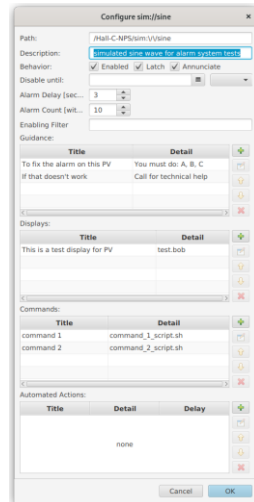
FIG.2. PV Alarm Configuration Menu
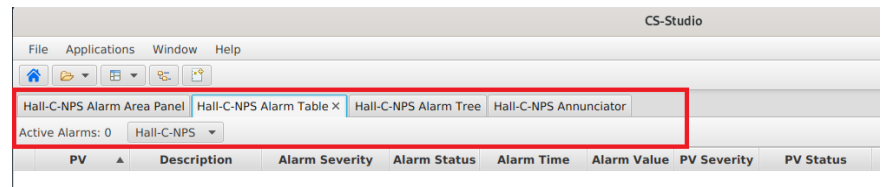
FIG.1. Phoebus Alarm System Programs

FIG.3. Custom Phoebus Alarm System User Interface for Hall C NPS