



Atacama Large Millimeter / submillimeter Array

Alarms Configuration GUI

ALMA-BBB-CCC-A-DDD

Version: A
Status: Draft

2010-05-24

Keywords: alarm system

Prepared	Tomás Staig	xx/xx/2010	
	Name	Date	Signature
		
	Rodrigo Tobar	xx/xx/2010	
	Name	Date	Signature
		
	Cristián Maureira	xx/xx/2010	
	Name	Date	Signature
		
Approved	---		
	Name	Date	Signature
		
Released	---		
	Name	Date	Signature
		

Change Record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
A	xx-05-2010	All	Creation.

Table of Contents

1	INTRODUCTION	1
1.1	Purpose	1
1.2	Scope	1
1.3	Reference Documents	1
1.4	Abbreviations and Acronyms	1
1.5	Stylistic Conventions	2
2	GENERAL OVERVIEW	2
3	DESIGN OVERVIEW	2
3.1	Implementation	2
3.2	Class Diagram	4
4	INSTALLATION MANUAL	5
5	USER MANUAL	6

1 INTRODUCTION

1.1 Purpose

In big projects, where a enormous number of variables and systems are related to each other, a simple action or accident can trigger a chain of reactions because of its nature. For example, if a network cable placed between two computers is cut, all applications using the network will fail, which may cause, in turn can cause other applications to crash, and so on. Then, the amount of errors that may appear in the system can obfuscate the root cause of its problematic state.

The ACS Alarm System (AS) is in charge of defining and report each of the failures of the system, and of reducing them when needed, so the user can track the root cause in the case of a problematic situation of the system.

Alas, the complete configuration of Alarm System (AS) is a complex task, since it consists in several XML files which relate each other. Because of this, the configuration process needs to be improved by means of using a user-friendly graphical tool.

The following document has been created to describe the usage of the Alarms Configuration GUI (ACG) tool, a Graphical User Interface (GUI) which allows to easily configure the Alarm System of ACS. In this document we provide a complete picture of the functionality of the software, along with some hints about its design.

1.2 Scope

This document is intended to be read by all the users of Alarm Configuration GUI (ACG), be them astronomers, operators, engineers or any other, to allow them to use the interface properly.

It is assumed that the reader has a good command of the Alarm System (AS) to understand every concept and process of this tool. If not, please refer to the Alarm System description document [1] for further details.

1.3 Reference Documents

[1] ALMA-SW-NNNN – ACS Alarm System, Revision 1.6

[2] Eclipse, *Building Commercial-Quality Plug-ins*, Second Edition, Clayberg, E. - Rubel, D.

1.4 Abbreviations and Acronyms

ACS	ALMA Common Software
AS	Alarm System
ACG	Alarms Configuration GUI
DAO	Data Abstraction Object

1.5 Stylistic Conventions

The following styles are used:

bold

In the text, to highlight words.

In the text, for commands, filenames, pre/suffixes as they have to be typed.

italic

In the text, for parts that have to be substituted with the real content before typing. Also used to highlight words or section names.

`teletype`

In the text, for commands, filenames, pre/suffixes as they have to be typed. Also used for file content examples.

2 GENERAL OVERVIEW

In many real-world situations, when a problem arises in certain object, it can trigger problems in other objects. For instance, a power supply failure can make an entire rack of computers fail; the failing computers would produce

The purpose of the ACS Alarm System (AS) is to let the user recognize, in a multiple failure scenario, the root problem of the situation. In the case cited above, the power supply failure triggered a great chain of events; nevertheless, this is the only reason why the system does not behave properly, and the other problems are caused (directly or indirectly) by it.

3 DESIGN OVERVIEW

3.1 Implementation

For more information please see the Javadoc in <http://csrg.inf.ut fsm.cl/acg/api/>.

DAOManager: Class responsible to deals with the complexity of the usage of the DAOs implemented in the `com.cosylab.acs.laser.dao` package. It handles an instance for each of the `DAOImpl` classes of the Alarm System. The other classes should retrieve the `DAOImpl` classes through this class.

AlarmManager: Class responsible to handling the references to the Alarm definitions that the ACG does.

AlarmSystemManager: This class representing the whole instance of the Alarm System configuration.

CategoryManager: Class used to manage all the information about categories that the ACG uses.

EntityManager: Common interface to all entities Managers.

ReductionManager: Class used to manage all the information about reductions.

SourceManager: Class responsible of handling all the references.

AcsInformation Class used to handle the connection of the application with the ACS Manager.

UserAuthenticationException: Exception class used by the authentication process of the user.

UserAuthenticator: This class is used to handle the authentication of the users that make use of the ACG. There are two kind of users:

- *Administrator:* The admin role allows the user to perform any action on the ACG. This includes: Sources CRM (Creation, Removal, Modification) Alarm CRM, Categories CRM and Reduction Rules CRM.
- *Operator:* The operator role should only make available to the user Alarms CRM in the ACG.

ReductionRule: Class used to represent a Reduction Rule.

IllegalOperationException: The IllegalOperationException exception should be used when illegal actions are going to be taken over a set of objects of the Alarm System configuration.



4 INSTALLATION MANUAL

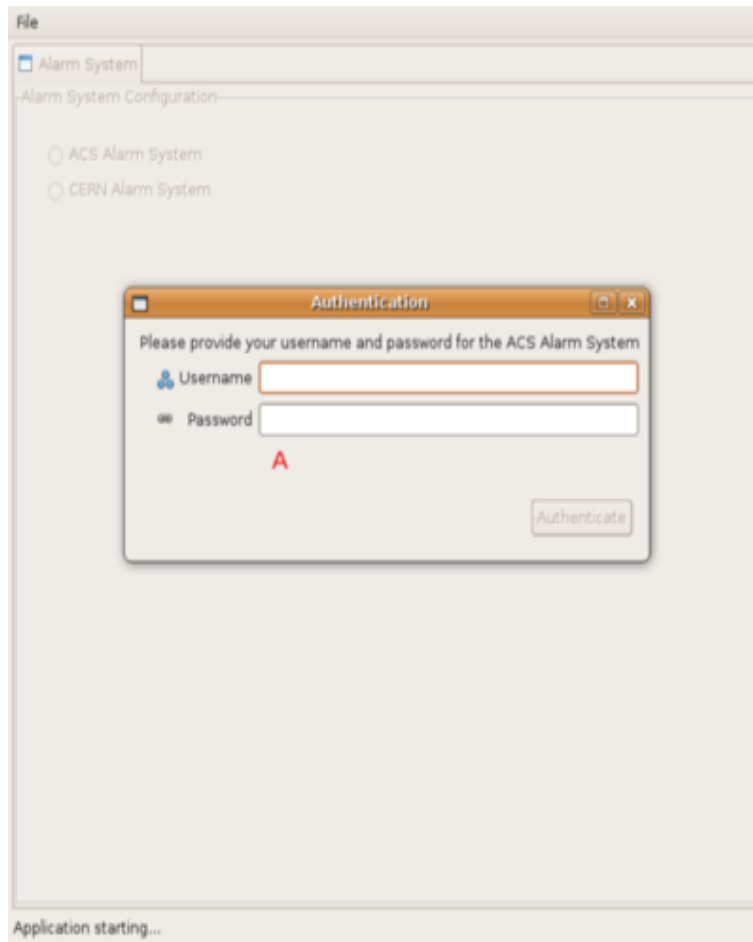
1. To install the **Alarms Configuration GUI / Alarms Definition GUI** you first have to acquire the *ACSLaser/alarmsDefGUI* module.
2. You need to have the *acsPluginBuilder* installed in order to obtain all the required ACS Jar files. If you need to install it, you can find this application in the *acsEclipseUtils* module.
3. Additionally you need to have the following jar files either in your **ACS** installation or in your *INTROOT*:

castor.jar	alarmssysteminterface.jar	jcont.jar
jacorb.jar	JavaContainerError.jar	acsjlog.jar
lasercore.jar	acsalarmSchemaBindings.jar	acserrj.jar
cdbErrType.jar	acsContainerServices.jar	junit.jar
maci.jar	maciSchemaBindings.jar	logging_idl.jar
jACSUtil.jar	ACSErrTypeCommon.jar	maciErrType.jar
logkit.jar	avalon-framework.jar	acserr.jar
acsASsources.jar	acsErrTypeAlarmSourceFactory.jar	cmwmom.jar
jms.jar	commons-logging.jar	xercesImpl.jar
cdbDAL.jar	log4j-1.2.8.jar	CDB.jar
cdbjDAL.jar		

4. All the files should be available in a standard installation of **ACS** but if the *acsPluginBuilder* application can't find any of those jar files it will fail to compile.
5. If you have all of the previous requirements, go to the *alarmsDefGUI/src* directory and execute the command `make`. After acquiring all the required jars it will start to compile the **ACG plugin**.
6. When the compilation is finished a zip file with all the required files will be placed in *alarmsDefGUI/object/headlessTemp/I.alarmsDefGUIBuild/* with the name *alarmsDefGUIBuild-linux.gtk.x86.zip*.
7. Use the command `unzip alarmsDefGUIBuild-linux.gtk.x86.zip`. Go to the extracted folder `cd alarmsDefGUI`.
8. Before executing the application edit the *acg.ini* file. Here you can set the manager and the *ACSROOT* and *JACORB.ROOT* arguments.
9. You need to set the *ACS_CDB* environment variable with a valid **CDB**. There is a valid **CDB** example in the *alarmsDefGUI/test/* directory. Now you can start **ACS** (*acsStart*) or update the **CDB** (*cdbjDALClearCache*) if **ACS** was already running.
10. To execute **ACG** just use the command `./acg`.

5 USER MANUAL

- When you start the application you will be prompted for a user name and password. After login an authorized user it will load the data from the CDB.



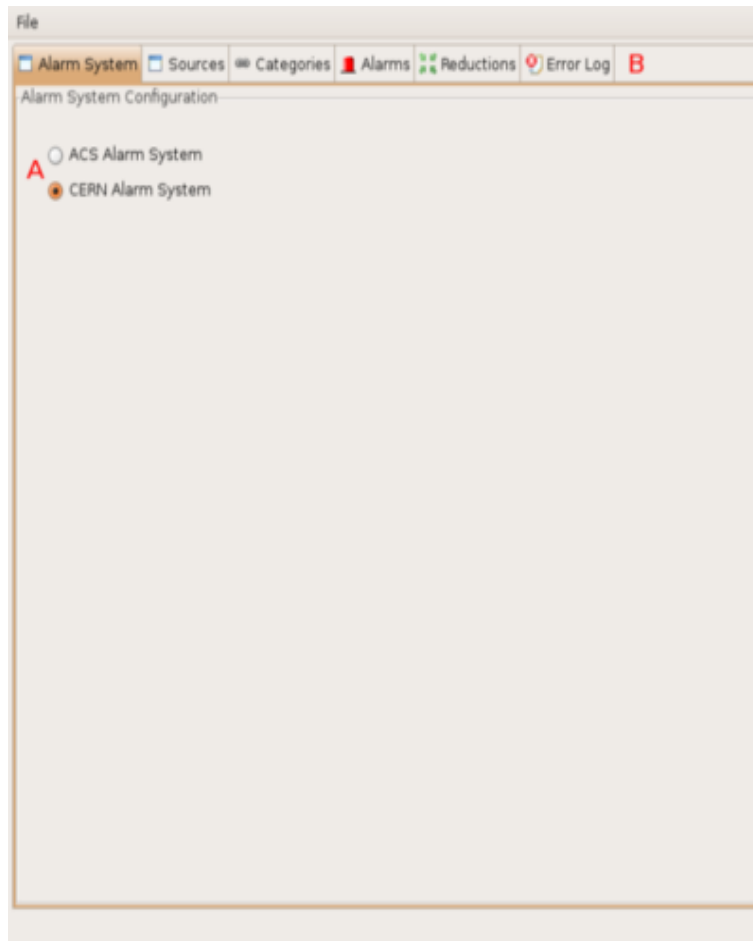
- **A:** Here you have to put the Username and Password *admin* and click the Authenticate Button or hit the Enter button.

- At any time after login you can access the File Menu. Here you can load data from the CDB and discard any change you have done so far. Save the current changes to the CDB or Exit the application.



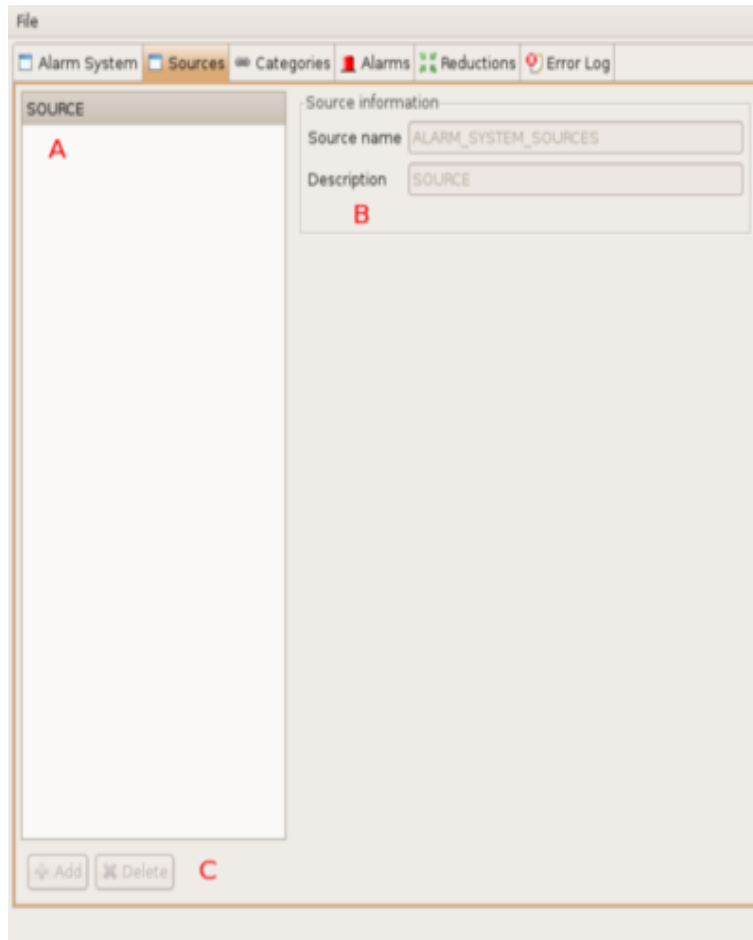
- **A:** Here you can choose to use the *Load CDB*, *Save CDB* and *Exit* actions from the *File Menu*.

- After loading the data from the CDB. The application will start in the Alarm System View.



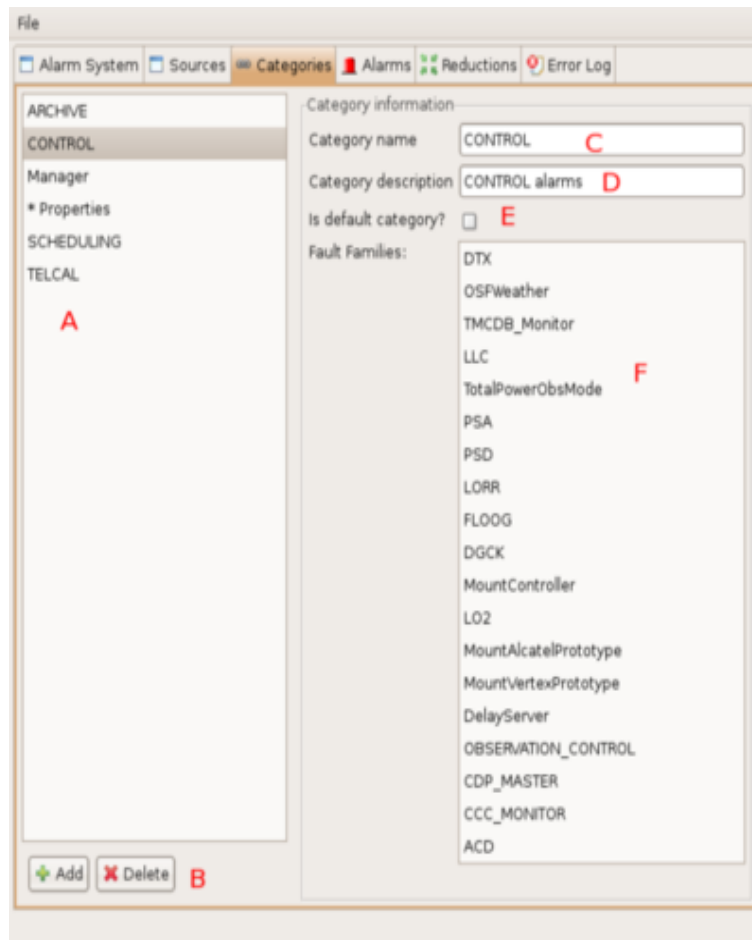
- **A:** Here you can choose between the ACS Alarm System Implementation or the CERN Alarm System Implementation. For this purpose you have to select the corresponding Radio Button. If ACS Alarm System is chosen, all the Tabs are disabled, since current ACS Alarm System implementation is just a logger.
- **B:** This is the Tab Selection Panel. When you click any of the tabs it will go to the chosen section.

- This is the Sources view, here you will be able to edit Source information. Currently all the editing is disabled in this view since ACS only supports one Source for the Alarm System.



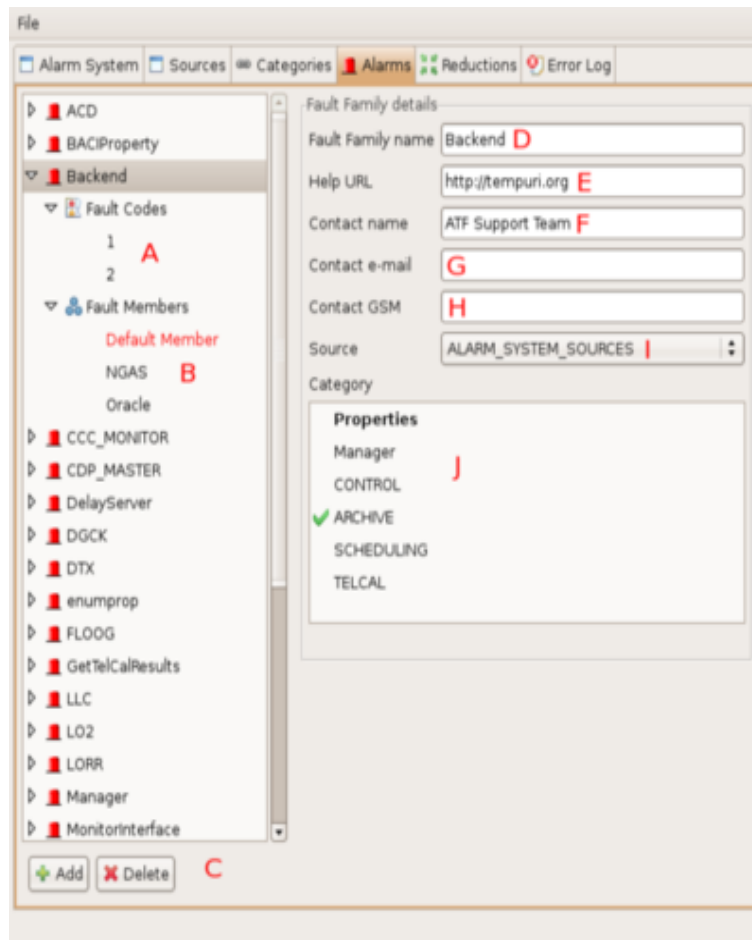
- **A:** This is the list of available sources. You should click the source you wish to edit.
- **B:** Here you can modify the Source Name and Description.
- **C:** When you have a selected source in the list, you can delete it by clicking in the Delete Button. You could also add new Sources by clicking the Add button.

- This is the Categories View, here you will be able to edit Category information and add or remove Fault Families from the categories.



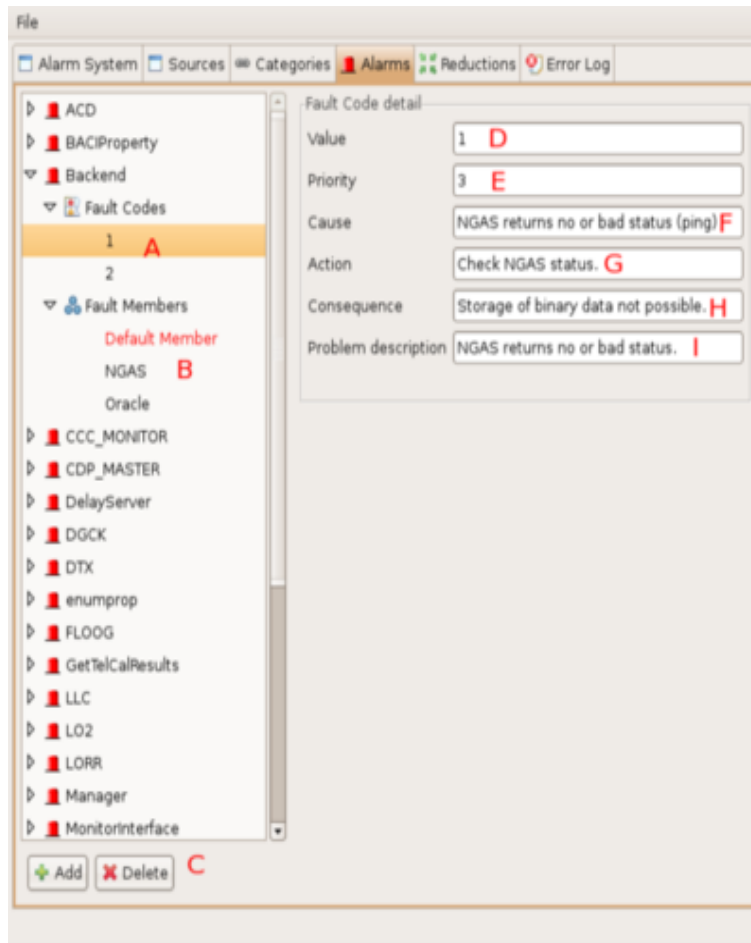
- **A:** This is the list of available categories. You should click the category you wish to edit. You can also access a new menu by right clicking in the list. This menu can do the same as described in B, but by clicking the corresponding actions.
- **B:** If you have a selected category you can delete it by clicking in the Delete button. You could also add new Categories by clicking the Add button.
- **C:** Here you can edit the Category Name by writing the desired Name.
- **D:** Here you can edit the Category Description by writing the desired description.
- **E:** By clicking the Check Box you could set the current Category as the default one. It is important to notice that there could only be one default Category. If you choose a new default Category the previous one will be unset.
- **F:** You can access a new menu by right clicking the Fault Family List. If you have selected a Fault Family, you can remove it by clicking the corresponding action. You can also edit the list of Fault Families by clicking the Add a New Fault Family action.

- This is the Alarms View. Here you will be able to edit the Alarms information, it is separated in Fault Families, Fault Members and Fault Codes.



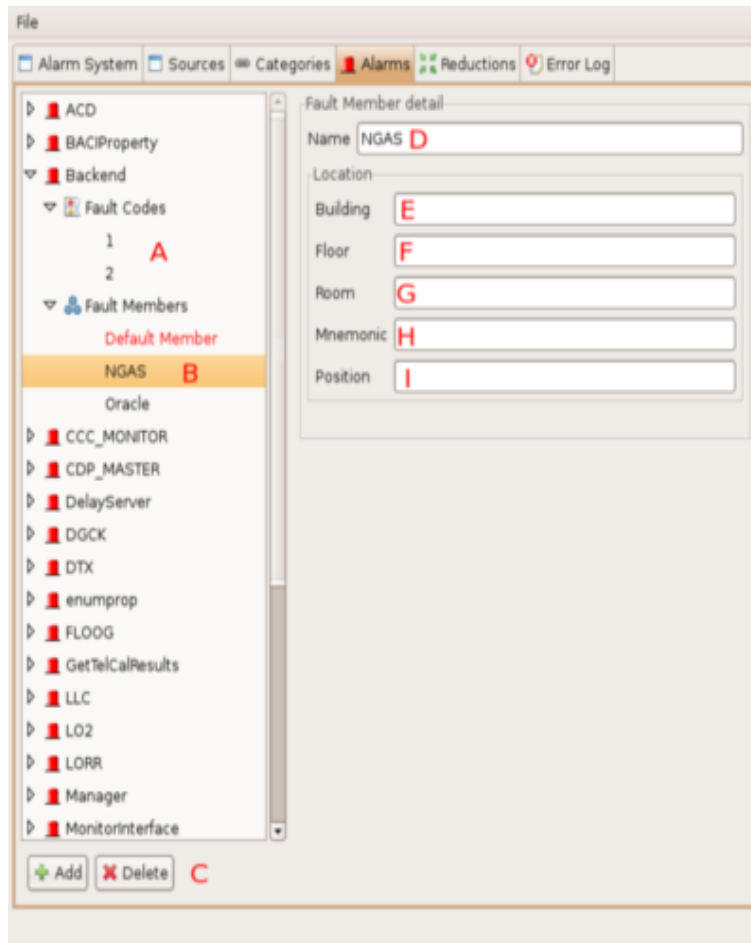
- **A:** Here you can select a FaultCode. You can right click to add a Fault Member or Fault Code
- **B:** Here you can select a FaultMember. You can right click to add a Fault Member or Fault Code
- **C:** Here you can delete the selected element or add a Fault Family.
- **D:** You can edit the Fault Family name.
- **E:** You can edit the Help URL.
- **F:** You can edit the Contact name.
- **G:** You can edit the Contact E-Mail.
- **H:** You can edit the Contact GSM.
- **I:** You can select the Source.
- **J:** This is a List of Categories, you can add or remove one by double clicking or using the space key.

- This is the same Alarms View with a Fault Code selected.



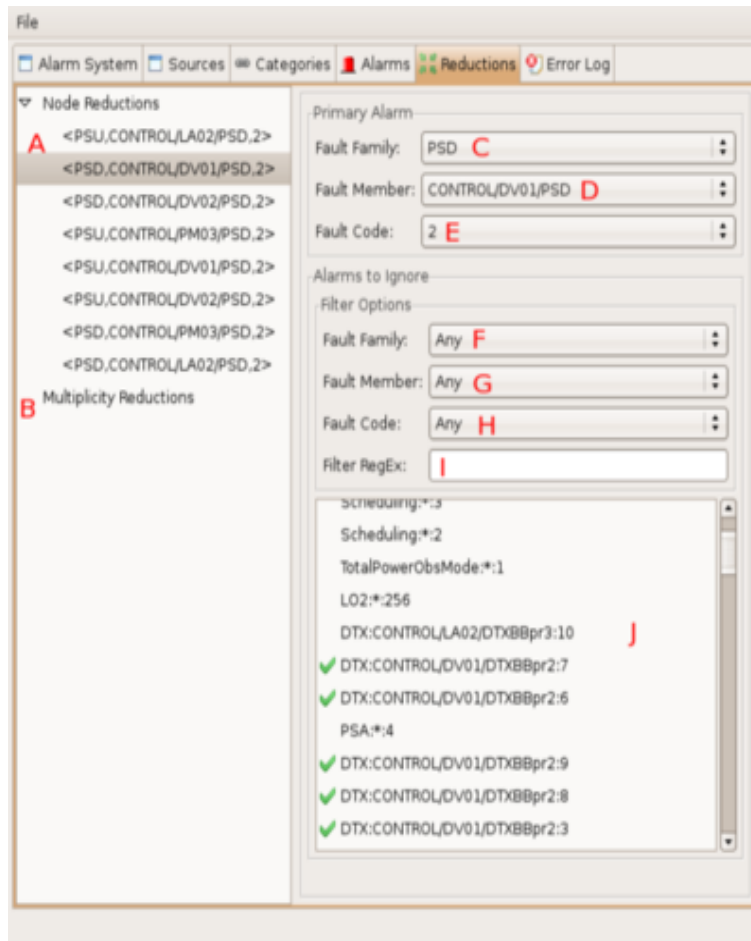
- **A:** Here you can select a Fault Code. You can right click to add a Fault Member or Fault Code.
- **B:** Here you can select a FaultMember. You can right click to add a Fault Member or Fault Code.
- **C:** Here you can delete the selected element or add a Fault Family.
- **D:** Here you can edit the Fault Code Value.
- **E:** Here you can edit the Priority.
- **F:** Here you can edit the Cause.
- **G:** Here you can edit the Action.
- **H:** Here you can edit the Consequence.
- **I:** Here you can edit the Problem Description.

- This is the same Alarms View with a Fault Member selected.



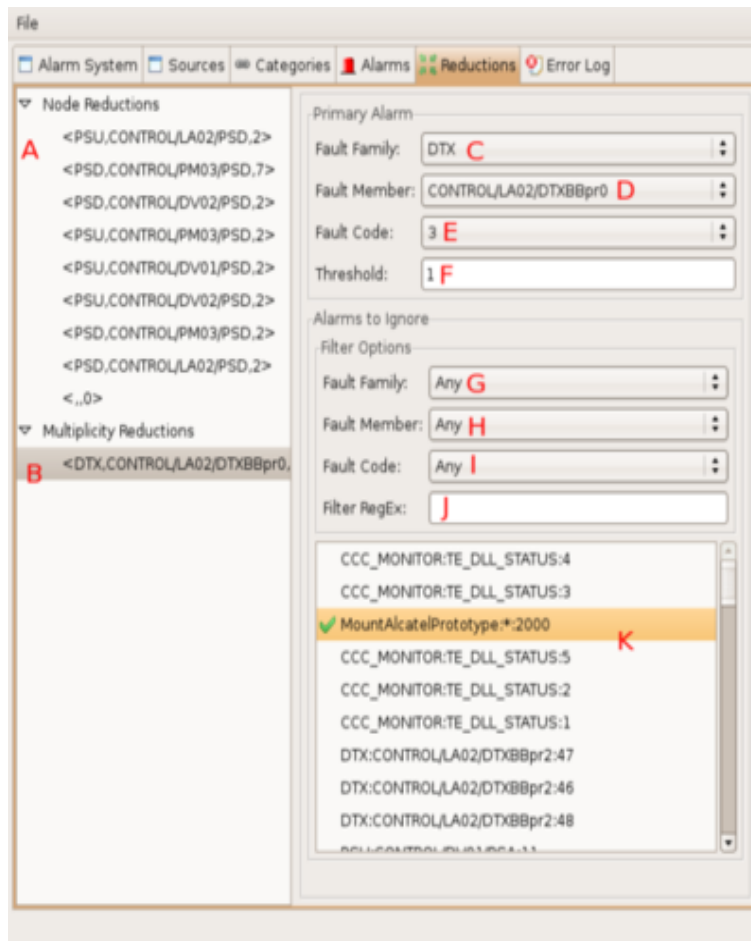
- **A:** Here you can select a FaultCode. You can right click to add a Fault Member or Fault Code.
- **B:** Here you can select a FaultMember. You can right click to add a Fault Member or Fault Code.
- **C:** Here you can delete the selected element or add a Fault Family.
- **D:** Here you can edit the Fault Member Name.
- **E:** Here you can edit the Building.
- **F:** Here you can edit the Floor.
- **G:** Here you can edit the Room.
- **H:** Here you can edit the Mnemonic.
- **I:** Here you can edit the Position.

- This is the Reduction Rules View. Here you can edit the Node and Multiplicity Reduction Rules. A node Reduction Rule is selected.



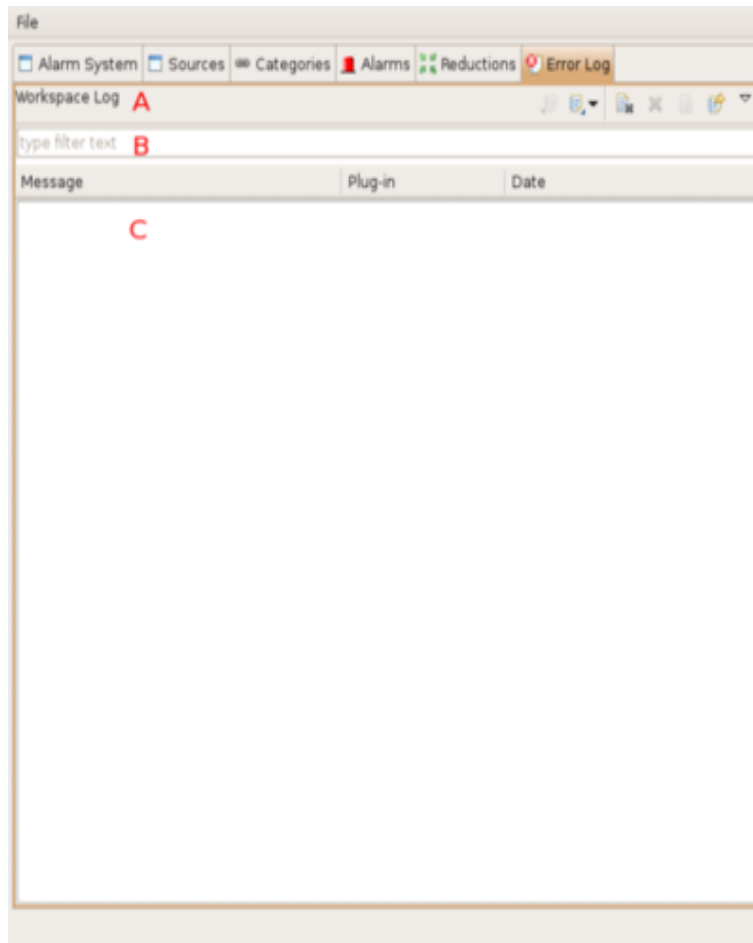
- **A:** Here you can select a Node Reduction Rule.
- **B:** Here you can select a Multiplicity Reduction Rule.
- **C:** You can select the parent FaultFamily.
- **D:** You can select the parent FaultMember.
- **E:** You can select the parent FaultCode.
- **F:** You can select the FaultFamily to show in the Alarm List.
- **G:** You can select the FaultMember to show in the Alarm List.
- **H:** You can select the FaultCode to show in the Alarm List.
- **I:** Here you can Filter the Alarm List with a given Regular Expression.
- **J:** This is the Alarm List of Child candidates. You can double click or press the space key to change the state of a child.

- This is the same Reduction Rules View but with a Multiplicity Reduction Rule selected.



- **A:** Here you can select a Node Reduction Rule.
- **B:** Here you can select a Multiplicity Reduction Rule.
- **C:** You can select the parent FaultFamily.
- **D:** You can select the parent FaultMember.
- **E:** You can select the parent FaultCode.
- **F:** You have to set the Threshold here.
- **G:** You can select the FaultFamily to show in the Alarm List.
- **H:** You can select the FaultMember to show in the Alarm List.
- **I:** You can select the FaultCode to show in the Alarm List.
- **J:** Here you can Filter the Alarm List with a given Regular Expression.
- **K:** This is the Alarm List of Child candidates. You can double click or press the space key to change the state of a child.

- This is the Error Log View. You will be brought to this View whenever an unhandled exception is found. If this were to happen, it would be appreciated to report the corresponding bug attaching the detailed information as mentioned in C as well as a way of reproducing the error if known.



- **A:** This is a Toolbar that allows you to import/export logs, clear the log for the session or delete it, as well as some other options.
- **B:** This Text Box help you find a specific Error in the list by filtering them.
- **C:** This is a list of unhandled exceptions that have occurred within this application. If you double click an error you can see detailed information about it.