



SAE – Automation - Interoperability for your devices and software applications.

Alarms in OpcDbGateway

Functionality of the proprietary alarm system

Alarm Systems in OpcDbGateway

Two alarm systems are implemented in [OpcDbGateway](#):

1. Alarm system implemented according to the specification OPC AE (Figure 3 – the directory „Alarms for OPC AE protocol“)
2. Proprietary **alarm system** (Figure 3 – the directory „Alarms (for Memory operands)“)

The functionality of both alarm systems can be configured using configuration application of [OpcDbGateway](#). It is assumed that the user has own alarm client implemented according to OPC AE specification. The second type of **alarm system** has several advantages:

- It provides recording of alarm history (history of transitions between states of alarm sources)
- Continuous status update of each **alarm source** is provided in the **alarm status table** in process database.
- It includes **alarm client** in the OpcDbGateway configurator
- There is an alarm **web client** available as well.
- User can create own alarm client according to the defined interface given by system OPC items and the structure of alarm tables in process database.,
- Using the application **SAEAUT SMS Service** delivered with OpcDbGateway **alarm messages** can be sent **by SMS and / or E-mail**.

Alarms in proprietary alarm system

The source of alarm in the proprietary alarm system of OpcDbGateway is memory *operand* (MO). Variables (data points) of variety external data sources such as OPC servers, other communication drivers of different devices and cells of database tables can be mapped to MO's in OpcDbGateway. MO can contain results of various logical operations (functions) with variables saved in other MO's. Because of this, alarm generated by OpcDbGateway can depend on **evaluation of different easy or complex conditions** in external devices or connected software applications.

Whether MO is to be a source of alarm or not is defined by its configuring. MO to become a source of alarm, it must be assigned a preconfigured *definition of alarm processing*. If the definition is not assigned, then the MO cannot be the source of an alarm. What contains the definition of alarm processing will be explained later.¹

¹ If we compare this with alarm systems according to OPC AE specification, the difference is that the source of the alarm is here OPC variable of the internal OPC DA server. (Of course, as MO can be mapped to OPC item as well, this claim of difference should be taken with caution.) The internal OPC DA server in OpcDbGateway is directly connected within OpcDbGateway with OPC AE server, which contains its own OPC DA client and OPC AE server interface for an external OPC AE client.

Static and dynamic alarming

Static alarming differs from **dynamic** that it **does not / does** validate (acknowledge) status of the alarm state by human operators. The *acknowledgment source* is again MO in OpcDbGateway. Therefore, similar to the *alarm source*, it can be the result of evaluating of a complex operation executed using several MO's.

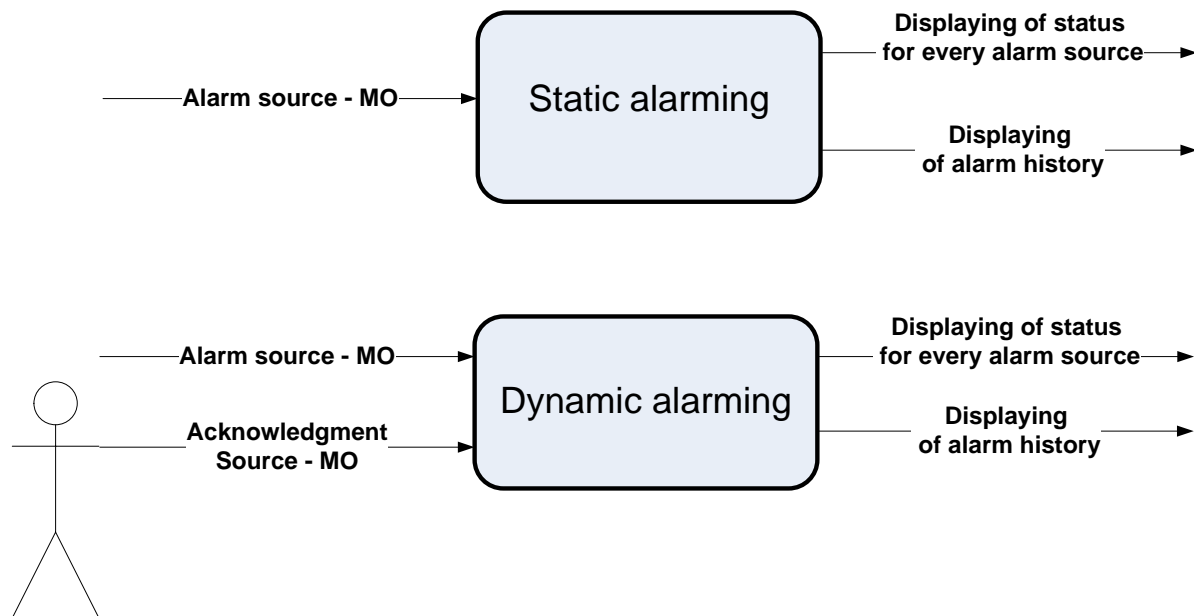


Figure 1 Static / dynamic alarming

States and transitions between states of alarms

For static alarming, each alarm source has **2 states**:

- Activated (come)
- Inactivated (not active or gone)

It is due to the fact that the alarm is characterized only by a single variable - *alarm source*. In the **dynamic alarming**, the alarm entity is defined by two Boolean variables (MO's) - *alarm source* and *acknowledgment source*) representing a total of four possible states:

- Not active
- Activated (Come)
- Inactivated (Gone)
- Acknowledged

The alarms entities are visible in the alarm status table (in the configurator or in the web client) with a constant number of rows equal to the number of configured alarm sources. **Transitions between states** are recorded as events in **the table of alarm history**. Here is recorded when there was a transition from one alarm state to another as well as one of two possible **configurable messages**:

- Message when the value of the alarm source is changed from *false* to *true*,
- Message when alarm source value changes from *true* to *false*.

As mentioned above, the source of the alarm is MO. The alarm source has a definition of alarm processing (Figure 2, Figure 4), which includes:

- Alarming type - static / dynamic (item Reg. Ack – Dynamic, Figure-4),
- Alarm messages (in several languages) to be entered in the alarm history table at birth and at death alarm conditions,
- Whether the alarm is permitted to return from active to inactive status²,
- Source acknowledgment alarm (MO) - must be configured in advance,
- Whether on alarm activation should be sent to SMS and / or e-mail³.

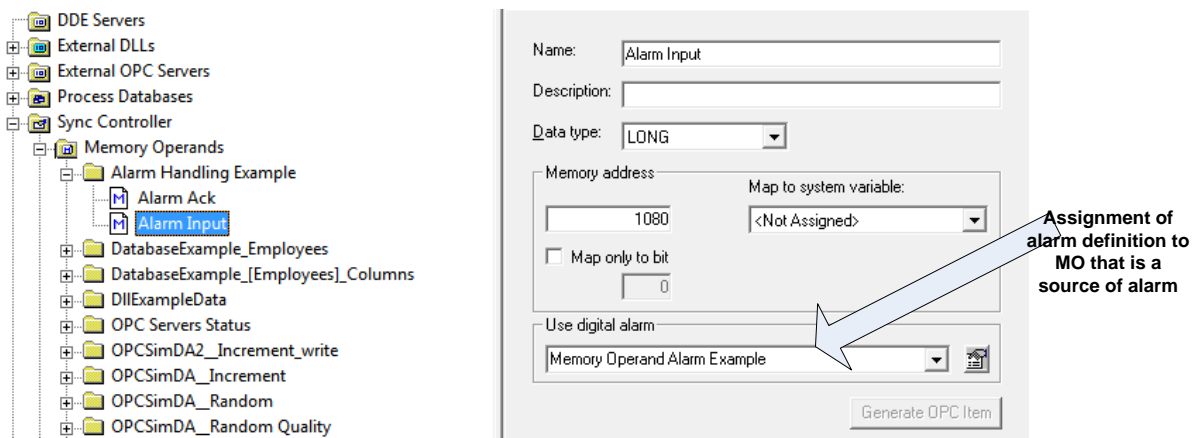


Figure 2 Assignment of alarm definition to the MO that is a source of alarm.

Multiple alarm sources (MO) can be assigned to the same definition of alarm processing. Several definition of alarm processing may contain the same definition of alarm message (Figure 5). Alarm messages can carry information about the values of selected variables (MO) at the time of start or end of alarm conditions (Figure 5)

Configure the alarm definition is carried out within [OpcDbGateway](#) configuration application under a tree structure labelled "Alarm handling (for memory operands)" - Figure 3.

² In case that it is enabled, it can execute transition from status "gone" to "Not active". If it is not enabled then it will be continually signalled that the alarm source has ever been in status "come".

³ Parameterization of the SMS and E-mail alarming is provided by configuration application of the SAEAUT SMS Service supplied with OpcDbGateway.

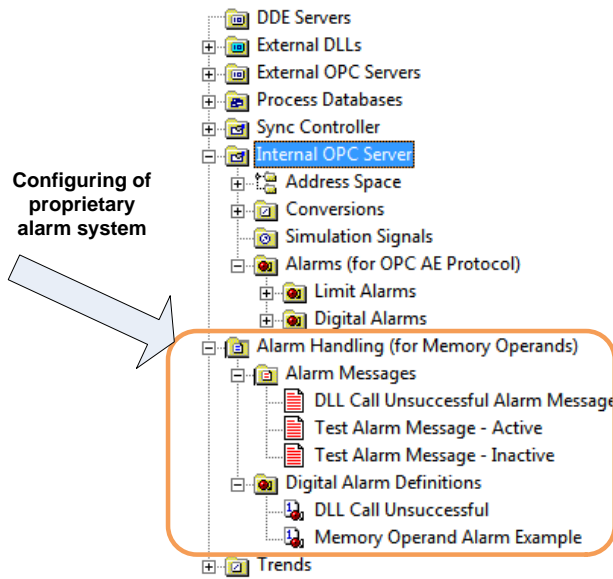


Figure 3 Configuring of alarm messages and alarm definitions for proprietary alarming

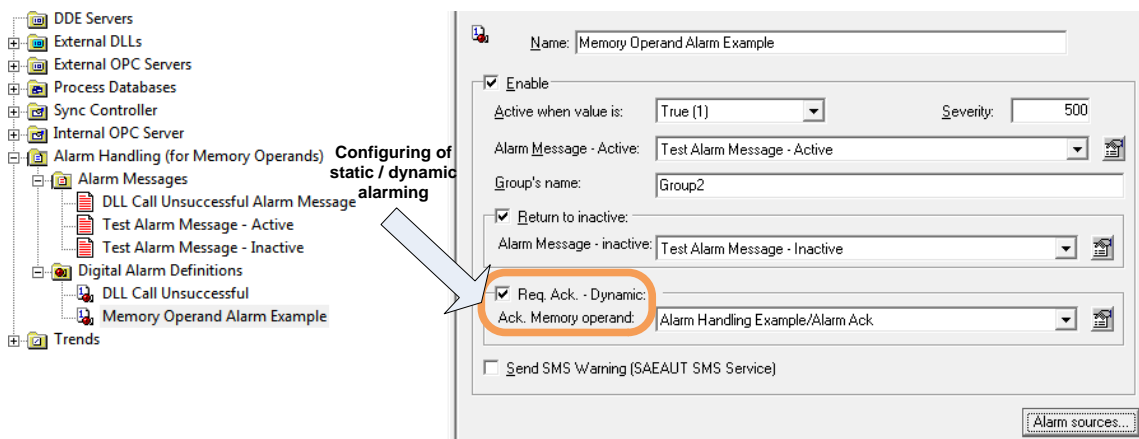


Figure 4 Alarm type configuring - static / dynamic

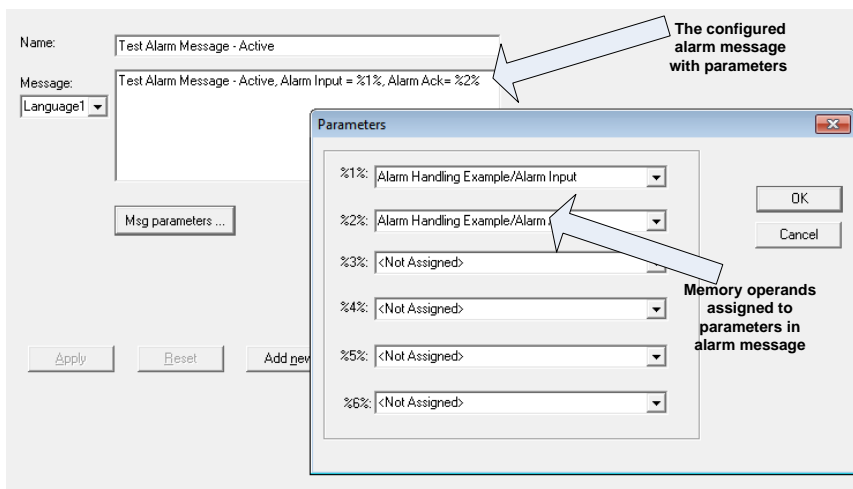


Figure 5 Configured alarm messages

Alarm client built-in configurator

Alarm client in the configurator is used for **debugging of alarm functionality for an application**. It can be displayed in a separate window "Output" initiated from the main configuration menu under "View". Within this view, the status table of alarm sources, table of alarm history and contents of the alarm log file can be displayed.

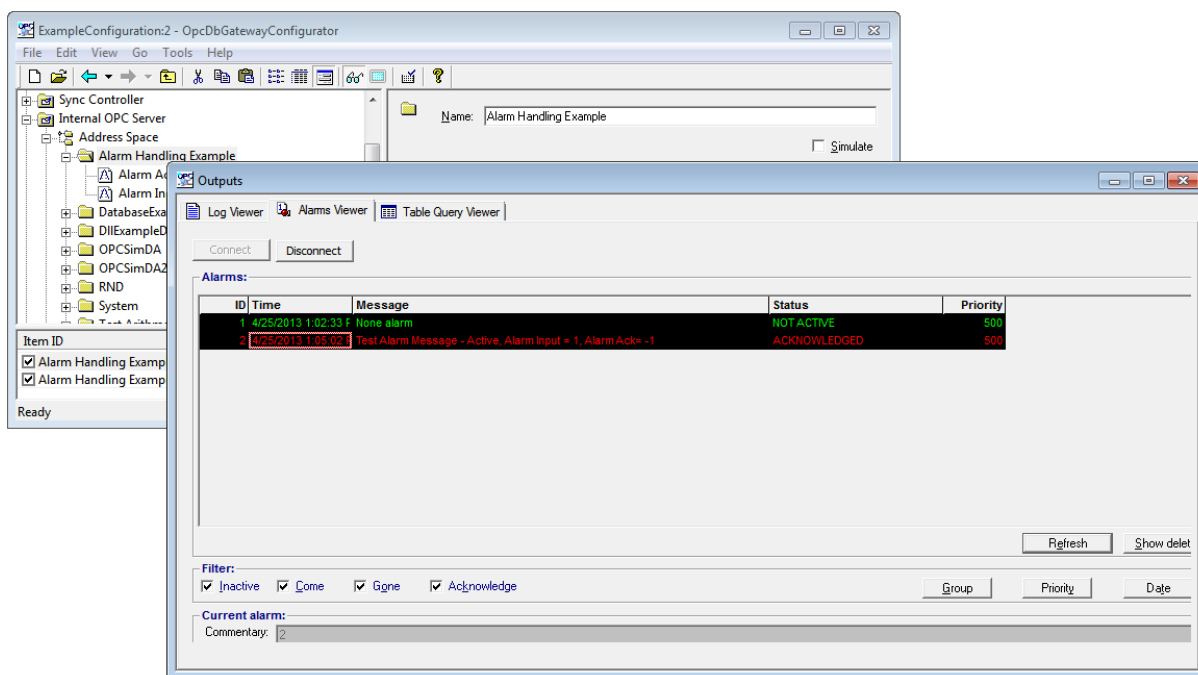


Figure 5 The "Output" view in configuration application to track the current status of alarm sources, the status transitions history and alarm log files.

Using the client different filters to display of alarm status can be chosen, acknowledged alarms and also a comment in the history of alarms for each alarm acknowledgment can be written.

Implementation of own client application to the alarm system

For specific applications it may be advantageous to create a **custom alarm client**. It must be implemented as an **OPC DA client** and at the same time as a **database client**. For management of alarm functionality, OPC variables are used hence there is need for OPC client functionality. The client also needs access to the alarm process database tables and hence there is the need for database client functionality. Internal OPC server of the OpcDbGateway offers OPC items to access the system variables (MO's) which can be used to **control the alarm functionality**. They can be found in the directory *System / Alarms*. One of these variables is also *ActualAlarmAck*. This variable can be used to acknowledge all alarm sources. This way all alarm sources would have common acknowledgment source. But, it is possible to configure alarms also such way that every alarm source will have own acknowledgment source.

Alarm history

As already mentioned, the history of transitions between states of alarm sources is entered in the table *AlarmStatusHistory* in the process database and also to the **alarm log file**. File size might increase over the acceptable size for long-term operation. This problem can be solved by generating new alarm log file using configurable event.

Summary

Proprietary alarm system [OpcDbGateway](#) has enough flexibility and many useful features. Using it to realize alarming for specific applications is relatively quick and easy. Of course, for those who have own OPC AE client or some specific requirements on alarming, [OpcDbGateway](#) provides other options based on its universal functionality of the interlinking data sources, external devices and software applications.

Copyright © 2013, SAE – Automation, s.r.o.

All rights reserved.

SAE – Automation and its logo are trademarks of SAE – Automation, s.r.o.

www.saeautom.sk, sae-automation@saeautom.sk