Phoenix

Alarm Forwarding

Installation and User Guide

Release 4 with Service Pack 1 (X.4.1.0)

ABM
DATA SYSTEMS, INC.

ISSUE	DATE	
1	May 2001	Initial release.
2	May 2003	Revised through X.4.0.
3	March 2004	Revised through X.4.1

ABM DATA SYSTEMS
9020 CAPITAL OF TEXAS HIGHWAY, SUITE 540
AUSTIN, TX 78759
(512) 345-6900

Alarm Forwarding Installation and User Guide

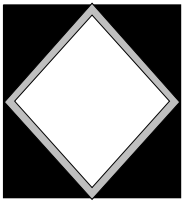


Table of Contents

How to Contact ABM Data Systems

ABM Data Systems, Inc.

9020 Capital of Texas Highway North, Suite 540
Austin, TX 78759
(512) 345-6900

Sales

ABM Sales hours are Monday through Friday, 8:00 a.m. to 5:00 p.m. Central Standard Time.

Sales Phone:	**(800) 767-7067**
Sales Fax:	**(512) 345-3252**
Sales Email:	**sales@abmdata.com**
World Wide Web Address:	**www.abmdata.com**

Technical Support

ABM Support hours are Monday through Friday, 8:00 a.m. to 5:00 p.m. Central Standard Time.

Support Phone:	**(800) 729-4226**
	(512) 345-6900
Support Fax:	**(512) 345-3252**
Support Email:	**support@abmdata.com**

This manual may help you resolve Phoenix problems. Please contact our Technical Support Department if we can help in any way.

Documentation Support

Please address problems, comments, or requests concerning documentation to **techdoc@abmdata.com**.

Alarm Forwarding—Overview

Alarm Forwarding is an add-on option to the Phoenix system that performs two functions, based on an Alarm Forwarding Sigtype:

- Deadman Monitoring: Phoenix notifies a Contact, such as a monitoring center supervisor, via pager, email, or fax when Pending Events have aged beyond a specified time without being processed by an operator.
- Auto Alarm: Phoenix processes and resolves Events by notifying Contacts via pager, email, or fax.

To notify by email, an email application such as MS Outlook, must be installed on the machine that will be notifying by email.

To notify by alpha or numeric pager, a modem must be connected to the machine that will be notifying by pager.

To notify by fax, Windows 2000 and a fax modem must be installed on the machine that will be notifying by fax.

Alarm Forwarding logs into Phoenix just like any other user. You can consider it a “virtual” Alarm Processing operator. On the machine that will be “working” Alarm Forwarding Events, Alarm Forwarding runs as a Service.

Installing Alarm Forwarding

You install Alarm Forwarding only on the machine that will be working Alarm Forwarding Events.

You may exit Setup at any time by clicking on the X in the top right corner.

1. On the machine that you are installing Alarm Forwarding, login to **Windows** under an account with administrative privileges.

2. Stop all Phoenix components that are running on the machine on which you are installing Alarm Forwarding. (If the Phoenix Application Server is running on another machine, you can leave it running.) If the installer finds components running, it displays a message and aborts the install.

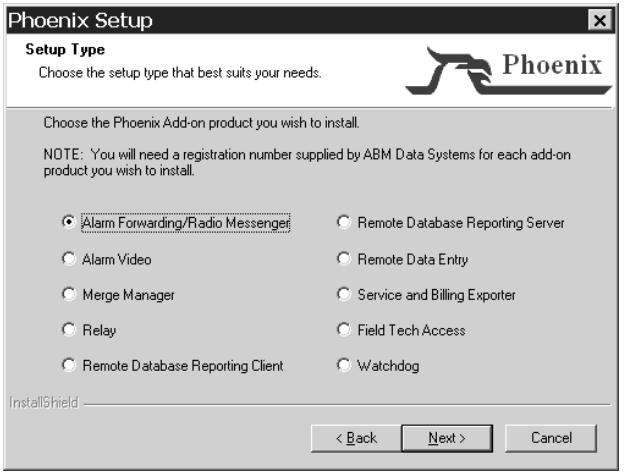
3. Insert the **Phoenix Release 3.4.1 CD** in the CD-ROM drive. Wait a few seconds for the Wizard to load.

   If the CD does not autostart: Open Windows Explorer, choose the CD drive and double-click on **setup.exe** in the disk directory.

4. Read the information in the **Welcome** window and click on the **Next** button.

5. On the **Select Components** window, choose **Install Add-on Components** and click on the **Next** button.

6. On the **Select Components** window, choose **Alarm Forwarding** and click on the **Next** button. (The components listed for Phoenix Desktop are different.)

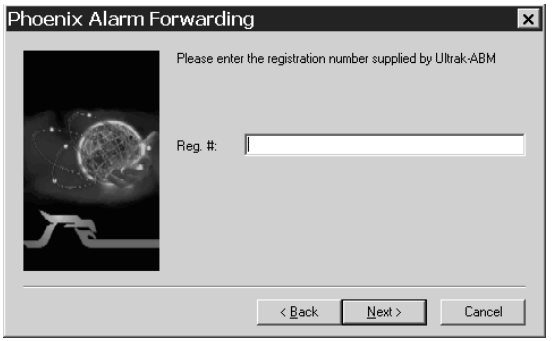


7. Read the information in the **Welcome** window and click on the **Next** button.

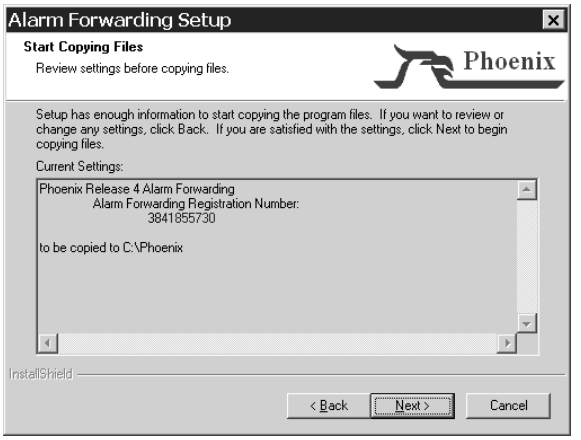
8. Carefully review the **License Agreement**. To continue with the install, click on the **Yes** button.

9. In the **Select Components** window, choose Alarm Forwarding if you will not use Radio Messaging. Choose both Alarm Forwarding and Radio Messaging if you will be using Radio Messaging.

10. In the **Phoenix Alarm Forwarding** window, enter the **Registration Number** that is provided on the CD sleeve.



11. In the **Start Copying Files** window, review the settings you have choosen, and if correct, click on the **Next** button. To make changes, click on the **Back** button.

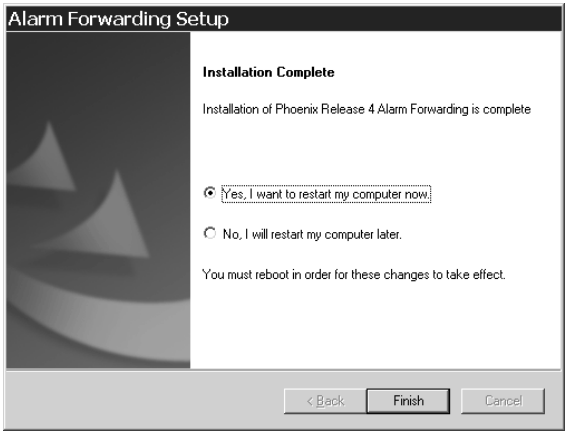


12. In the **Question** window, click on the **Yes** button if you want the Alarm Forwarding Service to start automatically when the computer is booted.

13. In the **Question** window, click on the **Yes** button if you want a start/stop Service shortcut.

14. In the **Question** window, click on the **Yes** button if you want the Email Service to start automatically.

15. In the **Installation Complete** window, click on the **Finish** button and reboot the computer.



16. If Alarm Forwarding will notify contacts using Email, the Alarm Forwarding Service must have Email capability.

a. Go to **Start** menu, **Control Panel**, **Services**.

b. In the **Services** window, highlight **PHOENIX Alarm Forwarding** and click on the **Startup** button.

c. In the **Log On As** area, select **This Account**, select a valid user (preferrably with a password that does not expire). Then click the **Add** button, and the **OK** button.

d. In the **Password** and **Confirm Password** fields, enter a valid password for the user, then the click **OK** button.

e. Close the **Services** window.

f. In the **Phx\tmp** folder, look for a folder named **mail**. If not present, create it.

Setting Up Auto Alarm

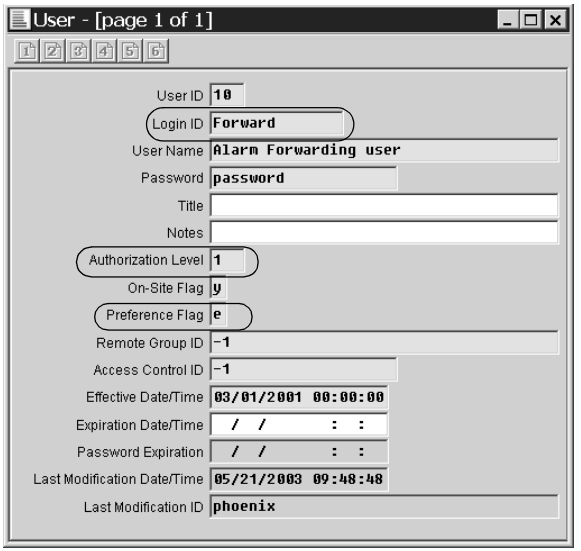
The Auto Alarm function of Alarm Forwarding enables Phoenix to automatically process and resolve Events by notifying the transmitter's Contacts via pager, email, or fax.

When an Event is generated that has a Sigtype designated for Alarm Forwarding, the Alarm Forwarding process selects the Event and attempts to send a message to the designated Contacts. If at least one Contact is notified without errors, the Event is resolved using the Resolution ID defined in the alarmforward.ini file. Alarm Forwarding can determine that it sent the message without errors, but it cannot determine if the message was actually received. ABM does not guarantee receipt of the message. If Alarm Forwarding is unable to notify at least one of the designated Contacts, the Event is returned to the Pending Queue, and a message is broadcast to all operators logged into Alarm Processing, notifying them of the new Event so that it may be processed manually.

To set up Auto Alarm, you must create records in several tables and update the alarmforward.ini file. Only the fields specific to Auto Alarm are discussed in this section.

1. In the **User** table, create a user that Alarm Forwarding will use to log in to Phoenix. You will enter this Login ID in the alarmforward.ini file (see Step 9 on page 13).
   a. In the **Login ID** field, enter a Login ID that Alarm Forwarding will use to login to Phoenix.
   b. In the **Password** field, enter **password**. (Alarm Forwarding ignores the password but you must enter one because it is a required field.)
   c. In the **Authorization Level** field, enter **1**.

Figure 1 *Record in User Table*



d. In the **Preference Flag** field, choose **e**, as shown in Figure 1.

2. In the **Class** table, create two records for use by Alarm Forwarding. One with a Type **sigtypes** and the second with a Type **instruction/contact**.

a. In the **Classifier ID** field for both records, enter a Classifier that starts with the **AA_**. For example, AA_Event, as shown in Figure 2.

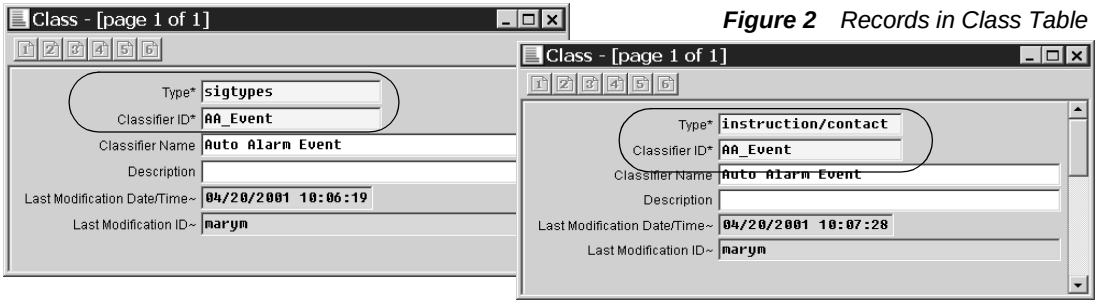


Figure 2 *Records in Class Table*

3. In the **Sigtypes** table, update or create a record for every Sigtype that will be handled by Auto Alarm. Phoenix uses the signal's Sigtype to identify signals that are to be processed by Auto Alarm.

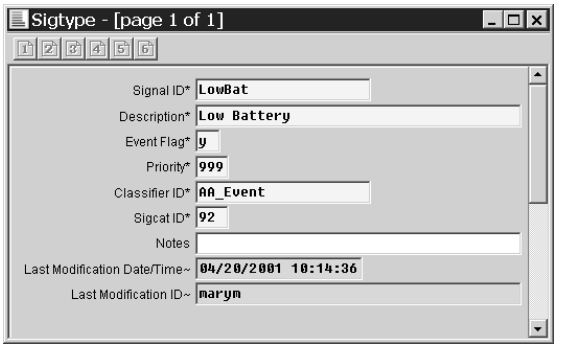


Figure 3 *Record in Sigtype Table*

a. In the **Priority** field, enter a Priority that is the lowest priority of all the Event-generating Sigtypes in the table.

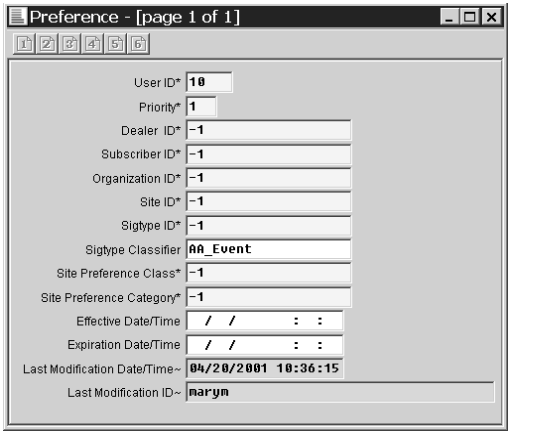
Only Alarm Forwarding Sigtypes may have this Priority number (except the marker record). You may need to change the priority of other Event-generating Sigtypes with a low priority.

Helpful Hint: To determine the lowest priority used in your system, open the **Sigtype** table, in the **Event Flag** field, enter a **y**, and in the **Priority** field, enter **999**,

then click on the **Query** tool. Click on the **List View** tool. Now you know which Event-generating Sigtypes in your system have a Priority of 999. If you have other Event-generating Sigtypes with a Priority of 999 in addition to the marker record and the Alarm Forwarding Sigtype, change the Priority of the other Sigtypes to a higher priority (lower number).

4. In the **Preference** table, create a record that prevents all users except Auto Alarm from selecting and processing Events that are designated for Auto Alarm.

Figure 4 *Record in Preference Table*



a. In the **User ID** field, enter the ID of the user you created for Alarm Forwarding to login with.

b. In the **Priority** field, enter **1**.

c. In the **hierarchy** fields, enter the marker value to apply Auto Alarm to all Events or enter a specific hierarchy level to limit Auto Alarm processing.

d. In the **Sigtype ID** field, enter **-1** for all Sigtypes, which will apply to every Sigtype with a Classifier ID starting with **AA_**.

e. in the **Sigtype Classifier** field, refer to the **Class** record with a Type **sigtypes** that you created in Step 2 and enter the value in the **Classifier ID** field. For example, AA_Event.

If your monitoring center uses Preferences for non-Auto Alarm signals, make sure you have a catchall Preference record that covers an Auto Alarm Event that has not been handled by Auto Alarm and must fall to an operator.

5. In the **Contacts Wizard**, create or update the **Contacts** for transmitters that send signals that will be processed by Alarm Forwarding.

   a. In the **Classifier ID** field, assign the appropriate Contacts the Alarm Forwarding Classifier that you created in Step 2.

   b. In the **Phones** fields, enter a pager number (alpha or numeric), an email address, or a fax number for each Contact

   For numeric pager numbers, enter commas for pauses as needed.

   For alpha pager, you must have an email accessed pager. For example, ATT World Net Paging. Enter the number in the email address For example, 555-2222@att.net.

   The Contact may also contain regular phone numbers if it is also used for non-Alarm Forwarding Events (has a different Classifier in Contact Links).

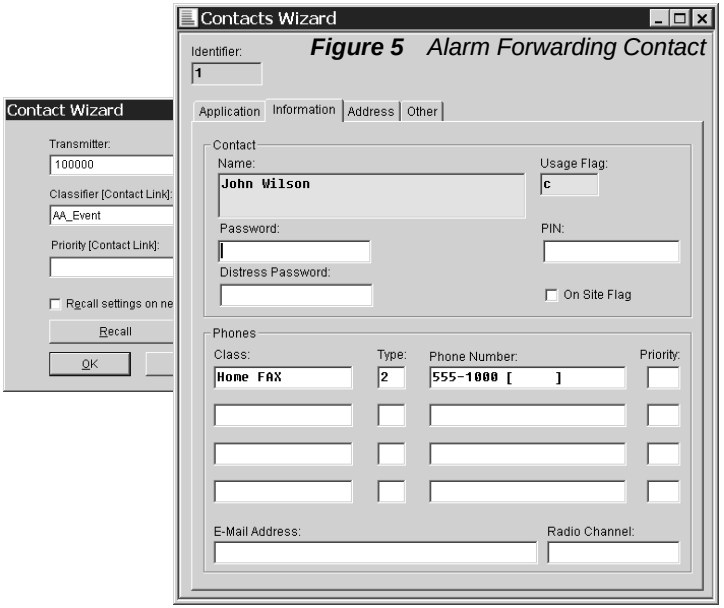
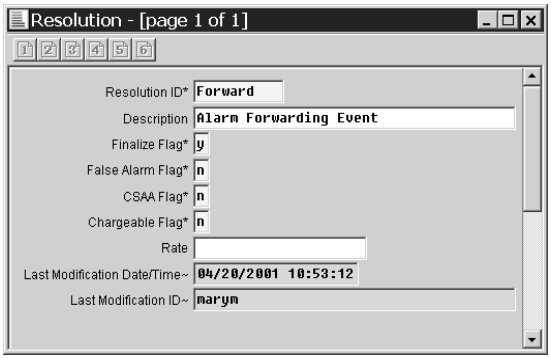


Figure 5 *Alarm Forwarding Contact*

6. In the **Resolution** table, create a record that Alarm Forwarding will use when it resolves an Event.

   Figure 6 *Resolution Record*

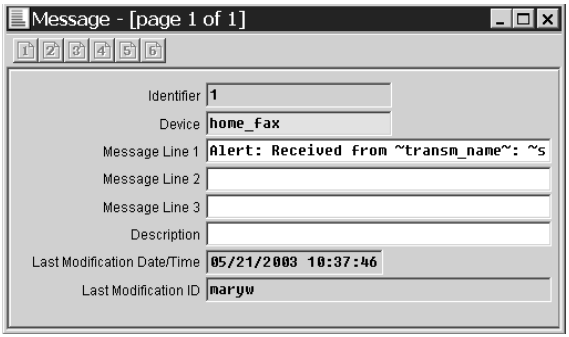
   a. In the **Resolution ID** field, enter a Resolution that clearly defines it as resolved by Alarm Forwarding.

   b. In the **Description** field, enter a more detailed description of the ID. If this field is blank, Alarm Forwarding uses **Finalized by Alarm Forwarding** for the description.

   c. In the **Finalize** field, enter **y** so the resolution may be used to resolve Events.

7. In the **Message** table, create one or more records that specify the message to send to the Contact and the notification method.

   Figure 7 *Message Record*

   a. In the **Device** field, choose the device that matches the Phone Class field in the Contact Wizard.

   b. In the **Message #** fields, enter the message you want Alarm Forwarding to send to the Contact.

   You may use the variables listed below in the message. When it sends the message, Alarm Forwarding will translate the variables to actual values based on the specific Event.

For example:

Message Field in Message Table	Actual Email or Fax Message
ALERT: Received from ~trans_name~: ~sigtype~ on zone ~zone_id~ - ~zone~.	ALERT: Received from Cox Office Supply: low battery on zone 10 - front office smoke detector

Type the variables in lower case only.

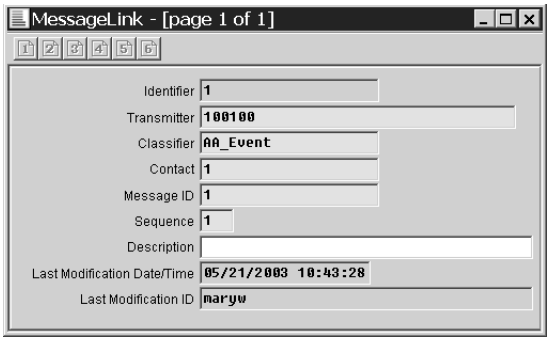
Variable	Alarm Forwarding uses the value from
~sigtype~	Description field from Sigtype table
~location~	Address 1, Address 2, Address 3, City, State fields in the Transmitter table
~tran_name~	Location field from the Transmitter table
~zone_id~	Zone ID field from Zone table
~zone~	Zone Name from Zone table
~trans_addr1~	Address 1 field in the Transmitter table
~trans_addr2~	Address 2 field in the Transmitter table
~trans_addr3~	Address 3 field in the Transmitter table
~trans_city~	City field in the Transmitter table
~trans_state~	State field in the Transmitter table
~trans_zip~	Zip field in the Transmitter table
~trans_country~	Country field in the Transmitter table
~dealer~	Dealer Name field in the Dealer table
~site~	Site Name field in the Site table
~site_addr1~	Address 1 field in the Site table
~site_addr2~	Address 2 field in the Site table
~site_addr3~	Address 3 field in the Site table
~site_city~	City field in the Site table
~site_state~	State field in the Site table
~site_zip~	Zip field in the Site table
~site_country~	Country field in the Site table
~signal_create_time~	Signal Create Date/Time field in the Signal table
~trans_time~	Transmitter’s Date/Time field in the Signal table

~relatedinfo~	Related Information field in the Signal table
~contact_name~	Name field in the Contact table
~user~	User Name field in the User table (Alarm Forwarding user)

c. In the **Description** field, enter any additional information that is relevant to the record.

8. In the **MessageLink** table, create a record that links a Transmitter and a Contact with the Message to send.

Figure 8 *Message Link Record*



a. In the **Transmitter** field, enter the Transmitter ID.

b. In the **Classifier** field, enter the Auto Alarm Classifier.

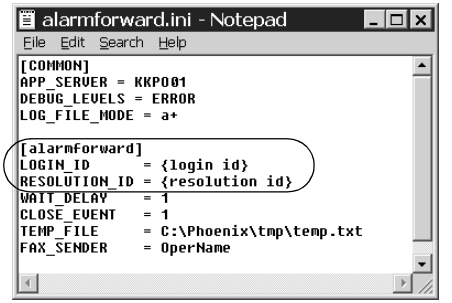
c. In the **Contact** field, enter the Auto Alarm Contact ID for the Transmitter.

d. In the **Message ID** field, enter the ID of the Message record in the Message table.

e. In the **Sequence** field, enter a number that indicates the order of the message if more than one message will be sent for the same notification.

9. Modify the **alarmforward.ini** file.

Figure 9 *Alarm Forward INI File*



a. In the **Login ID** parameter, enter the Login ID of the Alarm Forwarding user you created in Step 1 (remove the brackets).

b. In the **Resolution ID** parameter, enter the ID of the resolution you created in Step 6 (remove the brackets).

c. Save the changes to the INI file.

Setting Up DeadMan Monitoring

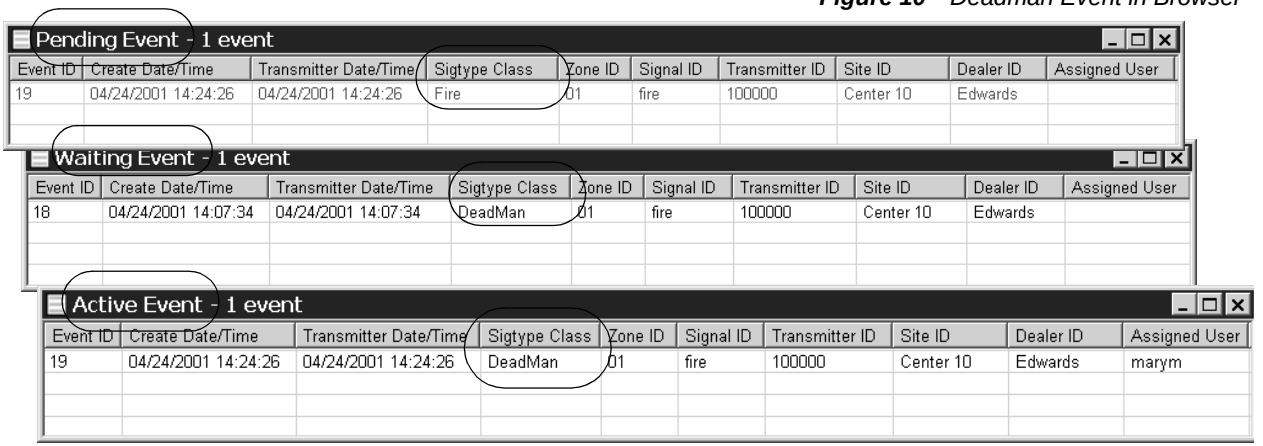
The Deadman Monitoring function of Alarm Forwarding notifies a Contact, such as a monitoring center supervisor, via pager, or email, or fax when any new Pending Event has aged beyond a specified time without being processed by an operator.

When the elapsed time expires, Deadman picks up the Event and notifies the contact. The Event is then put in the Wait Event Queue for the length of time defined by the user. When that time has elapsed, the Event is put back into the Pending Queue where it is selected by Deadman and the contact notified again.

Once an Event has been assigned to an operator (opened on the screen), Deadman will not pick up the Event again.

When an Event remains in Pending too long and Deadman notifies the Contact and moves the Event to the Waiting Event Queue, the Event's Sigtype Class in Browser changes to **Deadman**. When the Event is selected for processing by an operator and moves to the Active Event Queue, the DeadMan designation remains in the Sigtype Class field in Browser.

Figure 10 *Deadman Event in Browser*



To set up Deadman Monitoring, you must create records in several tables and update the appsrv.ini and the alarmforward.ini files. Only the fields specific to DeadMan Monitoring are discussed in this section.

1. In the **User** table, create a user that Alarm Forwarding will use to log in to Phoenix. You will enter this Login ID in the alarmforward.ini file (see Step 8 on page 19).

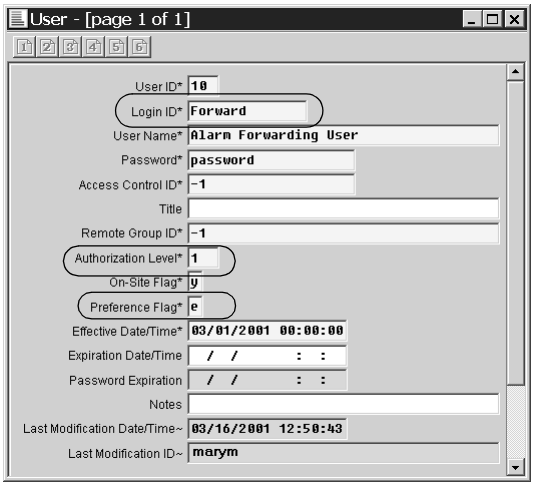
   a. In the **Login ID** field, enter a Login that you want Alarm Forwarding to use when it logins to Phoenix as a client.

   b. In the **Password** field, enter **password**. (Alarm Forwarding ignores the password, but you must enter one because it is a required field.)

   c. In the **Authorization Level** field, enter **1**.

   d. In the **Preference Flag** field, choose **e**, as shown in Figure 1.

Figure 11 *Record in User Table*



2. In the **Class** table, there is a record for use by Deadman, set up by the install/upgrade.

3. In the **Preference** table, create a record that is used by DeadMan Monitoring.

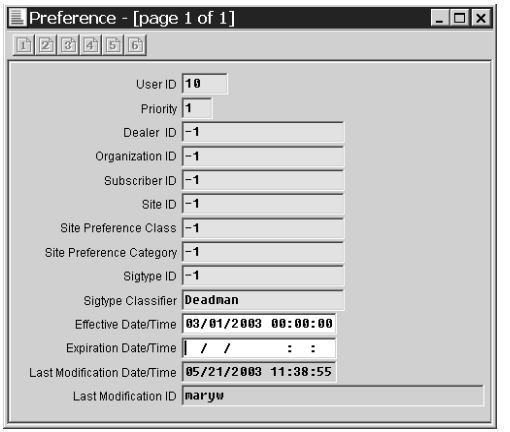
   a. In the **User ID** field, enter the ID of the user you created for Alarm Forwarding to login with.

   b. In the **Priority** field, enter **1**.

   c. In each **hierarchy** field, enter the marker value to apply Deadman to all Events.

   d. In the **Site Preference Class** field, choose **-1**.

Figure 12 *Deadman Record in Preferences Table*



e. In the **Site Preference Category** field, choose **-1**.

f. In the **Sigtype ID** field, choose **-1** for all Sigtypes.

g. in the **Sigtype Classifier** field, choose **DeadMan** from the dropdown list.

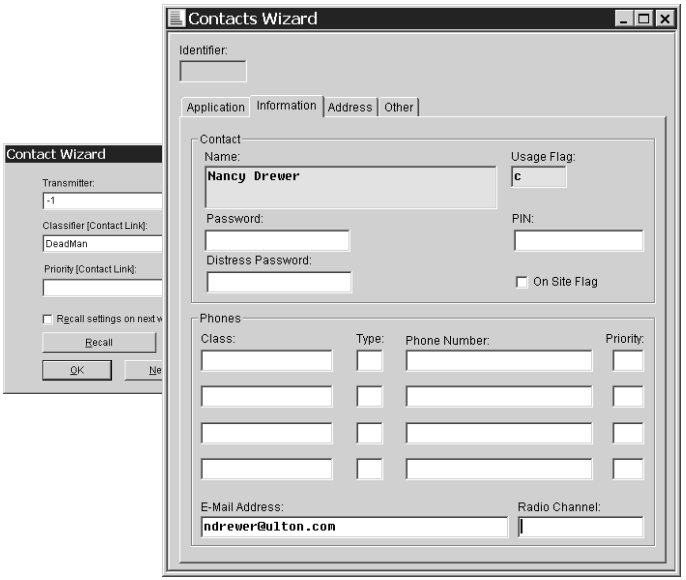
4. Create or update the **Contacts** to be notified when alarms are not processed within the specified amount of time.

   a. In the **Transmitter ID** field, choose the marker value, **-1**, so that it applies to all transmitters.

   b. In the **Phones** fields, enter a pager number, an email address, or a fax number for the Contact.

      For numeric pager numbers, enter commas for pauses as needed.

Figure 13 *Creating a Contact for Deadman Monitoring*

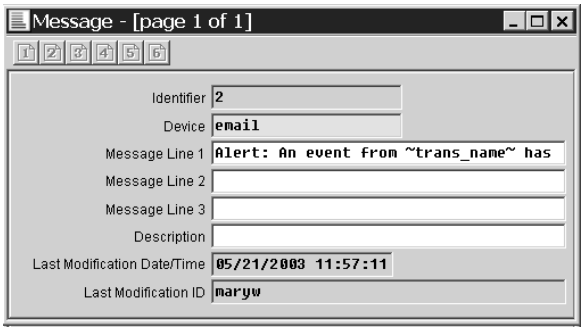


For alpha pager, you must have an email accessed pager. For example, ATT World Net Paging. Enter the number in the email address For example, 555-2222@att.net.

The Contact may also contain regular phone numbers if is it also used for non-Alarm Forwarding Events (has a different Classifier in Contact Links).

5. Create records in the **Message** table that specify the notification method and the message to send to the Contact.

Figure 14 *Message Record*



a. In the **Device** field, choose the device that matches the Phone Class field in the Contact Wizard.

b. In the **Message #** fields, enter the message you want Alarm Forwarding to send to the Contact.

You may use the variables listed below in the message. When it sends the message, Alarm Forwarding will translate the variables to actual values based on the specific Event.

For example:

Message Field in Message Table	Actual Email or Fax Message
ALERT: An Event from ~trans_name~ has not been processed.	ALERT: An Event from Cox Office Supply has not been processed.

Type the variables in lower case only.

Variable	Alarm Forwarding uses the value from
~sigtype~	Description field from Sigtype table
~location~	Address 1, Address 2, Address 3, City, State fields in the Transmitter table
~tran_name~	Location field from the Transmitter table
~zone_id~	Zone ID field from Zone table
~zone~	Zone Name from Zone table
~trans_addr1~	Address 1 field in the Transmitter table
~trans_addr2~	Address 2 field in the Transmitter table
~trans_addr3~	Address 3 field in the Transmitter table

~trans_city~	City field in the Transmitter table
~trans_state~	State field in the Transmitter table
~trans_zip~	Zip field in the Transmitter table
~trans_country~	Country field in the Transmitter table
~dealer~	Dealer Name field in the Dealer table
~site~	Site Name field in the Site table
~site_addr1~	Address 1 field in the Site table
~site_addr2~	Address 2 field in the Site table
~site_addr3~	Address 3 field in the Site table
~site_city~	City field in the Site table
~site_state~	State field in the Site table
~site_zip~	Zip field in the Site table
~site_country~	Country field in the Site table
~signal_create_time~	Signal Create Date/Time field in the Signal table
~trans_time~	Transmitter's Date/Time field in the Signal table
~relatedinfo~	Related Information field in the Signal table
~contact_name~	Name field in the Contact table
~user~	User Name field in the User table (Alarm Forwarding user)

6. In the **MessageLinks** table, create a record that links a Transmitter and a Contact with the Message to send.

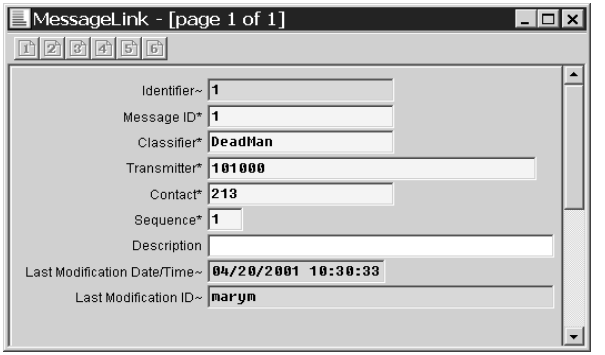
   a. In the **Message ID** field, enter the ID of the Message record in the Message table.

   b. In the **Classifier** field, enter the Auto Alarm Classifier.

   c. In the **Transmitter** field, enter **-1**.

   d. In the **Contact** field, enter the Deadman Contact ID.

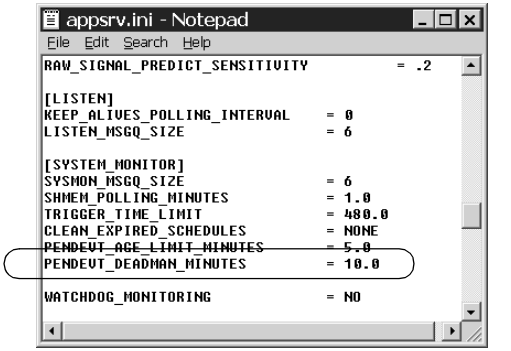
Figure 15 *Message Record*



e. In the **Sequence** field, enter a number that indicates the order of the message if more than one message will be sent for the same notification.

7. In the **appsrv.ini** file, enter a value in minutes in the PENDEVT_DEADMAN_MINUTES parameter. This value represents the amount of time, in minutes, that must elapse before an Event is picked up by Deadman processing. If this value is null, Deadman Monitoring does not function.

Figure 16 *appsrv.ini for Deadman Monitoring*

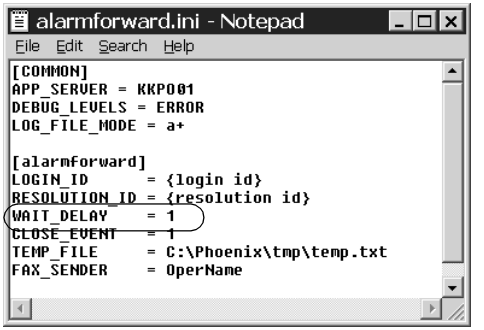


8. In the **alarmforward.ini** file, enter a value in the WAIT_DELAY parameter, which defines the length of time in minutes that the Event stays in the Wait Event Queue after Deadman processing notifies the Contact.

   In the CLOSE_EVENT parameter, enter the desired value: a value of **1** means the Event is sent only to Alarm Forwarding; **0** means the Event is sent to both Alarm Forwarding and an Operator.

   If the alarmforward.ini file does not exist in the profiles folder, you have not installed Alarm Forwarding. See “Installing Alarm Forwarding” on page 2.

Figure 17 *alarmforward.ini for Deadman Monitoring*



Using Alarm Forwarding

- Start the Alarm Forwarding Service on the computer.
- Start the Phoenix Mail Service on the computer. Go to Start menu, Phoenix, Mail.
- If the Phoenix Application Server Service is stopped, the Alarm Forwarding Service also stops.
- The History command in Alarm Processing indicates if an Event has been processed by Alarm Forwarding. For example, in Figure 18, the Action Log for Event 20 indicates this Event was processed by Alarm Forwarding.

Figure 18 *History in Alarm Processing*