



Administration

Creating Trigger Alerts

IMPORTANT NOTES

In the case of the death of a person (child, student or adult), One automatically generates a system warning that displays on the person's record. For more information, refer to *RG_Administration_System_Alerts*.

A number of predefined alerts are included in the One installation. Predefined alerts are always active and cannot be edited or deleted. For more information, refer to *RG_Administration_System_Alerts*.

Trigger Alerts

Alerts advise or warn users of changes in field values within a record. Alerts are displayed as an email, a popup message or a workflow message on the system administrator's **My Home Page**, or as a portal message for portal users. They are created if the data is persistent, such as a base closure or the death of a student. The trigger fires every time a user accesses the record.

Trigger alerts work on changes to nominated database tables and fields. The generic term for this type of alert is a database trigger. A trigger alert is based on field selection or SQL. This type of alert cannot operate on a list of fields and a free form SQL statement at the same time. Retrospective action can also be applied to a trigger alert.

A trigger alert requires a recipient to be added and an action to be defined before it can be activated. For more information, see the *Adding a Recipient* and *Defining an Action* sections.

A saved alert will not work until it is activated. For more information, see the *Activating an Alert* section.

Creating a Trigger Alert Using Field Selection

To create a new trigger alert using field selection:

1. Select **Tools | Administration | Alert Definition** to display the **Maintain Alert** page.
2. Click the **New** button to display the **Alert Definition [New Alert Definition]** page.

NOTE: The **Alert Definition ID** is a system generated number and cannot be changed.

3. From the **Alert Type** drop-down, select **Trigger**.
4. Enter a **Reason for Alert**.
5. Enter a **Description**.
An alert cannot be made active until it has been saved.
You cannot create a predefined alert; they are included in the One installation.
6. On the **Trigger Alert** panel, select an **Alert Based On** context. The **Database Trigger** and the table name are automatically populated.
7. Select the **Field Selection** tab to display all of the fields according to the context selected in the **Alert Based On** drop-down list.
8. Select the check boxes of the fields to be the alert subject.
9. If required, specify the parameters for **Only When Changed From** and **Only When Changed To**, e.g. **From True To False**.

You can view the SQL that One generates for this trigger via the **View Trigger SQL** panel.

10. Click the **Save** button.

The alert cannot be activated until a recipient has been added and an action has been defined.

Field	Alert When Changed	Only When Changed From	Only When Changed To
ADM NUM	<input checked="" type="checkbox"/>		
ASRGROUP	<input checked="" type="checkbox"/>		
BASE CLOSED	<input checked="" type="checkbox"/>		
BASE ID	<input checked="" type="checkbox"/>		

Where more than one field is selected, the alert will trigger only when all conditions are true. If you want the alert to trigger when any one of the conditions is true, you must create separate alerts.



Administration

Creating Trigger Alerts

Creating a Trigger Alert Using SQL

A trigger alert can be created using SQL. This type of alert cannot operate on a list of fields and a free form SQL statement at the same time.

To create a new trigger alert using SQL:

1. Complete steps 1 - 5 of *Creating a Trigger Alert Using Field Selection*.
2. On the **Trigger Alert** panel, select an **Alert Based On** context. The **Database Trigger** and the table name are automatically populated.
3. Select the **SQL** tab to display information on how to write the SQL.
4. Enter the SQL. This can only be a *Select* statement.
5. Click the **Verify SQL** button. The following message displays at the bottom of the screen: *SQL is valid*.

SQL is valid

2. Trigger Alert

Alert Based On: Base Details

Database Trigger: TRG_ALERT_A70B02154127

Field Selection | **SQL** | Retrospective Action

This should be a SQL SELECT statement that will be checked in the body of the TRIGGER using a CURSOR. If the number of rows returned by the CURSOR is greater than 0 then the TRIGGER will be fired.

Please note that the column values for the row being edited should be prefixed with the :OLD and/or :NEW in the SQL SELECT statement. Failure to do this will cause locking errors when the TRIGGER is fired.

If 'Retrospective Action' is applied for the Alert then existing rows satisfying the condition in the SQL SELECT statement will be affected. Please note that if the SQL uses :OLD and/or :NEW values then a replacement SQL for Retrospective Action must also be specified on the next page.

Verify SQL

6. Scroll down to the **View Trigger SQL** panel and click the **View Trigger SQL** button to view the SQL that is generated by One for this trigger.
7. Click the **Save** button. The alert cannot be activated until a recipient has been added and an action has been defined.

7. View Trigger SQL

Click to view the SQL that will be generated by this trigger

View Trigger SQL

```
CREATE OR REPLACE TRIGGER TRG_ALERT_A70B02154127
BEFORE INSERT OR UPDATE ON BASES
FOR EACH ROW
DECLARE LS_COMMAND CLOB;
CURSOR C_COND_CURSOR IS
select 1 from dual, addr_link al where old.address_id <> new.address_id and al.c_corresp = 'C';
DUMMY C_COND_CURSOR%ROWTYPE;
BEGIN
```

Creating a Trigger Alert Using Retrospective Action

Retrospective action can be used with field selection and SQL. However, it cannot operate on a list of fields and a freeform SQL statement at the same time. Retrospective action is only applied when the alert is activated.

To create a trigger alert using a retrospective action:

1. After the trigger alert has been created using field selection or SQL, select the **Retrospective Action** tab.
Alert Based On, **Database Trigger** and the table name are automatically populated.
2. Select the check box to retrospectively apply the conditions defined on the **Field Selection** tab or the **SQL** tab.
3. Click the **Verify SQL** button. The following message displays at the bottom of the screen: *Retrospective SQL is valid*.

Retrospective SQL is valid

2. Trigger Alert

Alert Based On: Base Details

Database Trigger: TRG_ALERT_A70B02154127

Field Selection | SQL | **Retrospective Action**

Tick the checkbox to retrospectively apply the conditions defined in the grid on the Field Selection tab page. Please note that this will trigger a Popup Message action (if one has been defined for the alert). It will only apply to those fields that have been selected in the grid on the Field Selection tab page or if free text SQL has been specified on the SQL will not apply if none or all of the fields are ticked. The action is only applied when the Alert is activated.

For e.g. if the SURNAME field has been selected in the grid on the Field Selection tab page and this checkbox is ticked then a Popup Message action (if one has been defined for the alert) will be raised for all existing PEOPLE records where the SURNAME field matches the criteria defined in the grid on the Field Selection tab page.

select 1 from dual, addr_link al where old.address_id <> new.address_id and al.c_corresp = 'C';

Verify SQL

4. If required, you can view the SQL that One generates for this trigger via the **View Trigger SQL** panel.
5. Click the **Save** button.

7. View Trigger SQL

Click to view the SQL that will be generated by this trigger

View Trigger SQL

```
CREATE OR REPLACE TRIGGER TRG_ALERT_A70B02154127
BEFORE INSERT OR UPDATE ON BASES
FOR EACH ROW
DECLARE LS_COMMAND CLOB;
CURSOR C_COND_CURSOR IS
select 1 from dual, addr_link al where old.address_id <> new.address_id and al.c_corresp = 'C';
DUMMY C_COND_CURSOR%ROWTYPE;
BEGIN
```



Administration

Creating Trigger Alerts

Adding a Recipient

All alerts require at least one recipient. The recipients and the actions (the way the alert is delivered) are selected in the **Recipient List** panel. The recipient **Type** determines the actions available for selection.

To add a recipient:

1. Click the required **Add** button to display the relevant browser or enquiry dialog.
Clicking the **Add Email** button displays an **Add email address** dialog for you to manually enter email addresses.
2. If required, use the search fields to narrow your search criteria.
3. Highlight a record and click the **Select** button to return to the **Recipient List** panel.
4. On the **Recipient List** panel, select the check boxes to send the alert by **Email**, **Popup** or **Workflow** message.

Type	Name	Description	Email Address	Email	Popup	Workflow	Portal Message	
Citizen		n/a		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
User	SYSUSER1	SYSUSER1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
User	SYSUSER3	SYSUSER3...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

- Add Users
- Add Portal Users
- Add Groups
- Add People
- Add Bases
- Add Posts
- Add By Query
- Add Email
- Remove

Please type the search criteria in the 'Looking For' to scroll to the item you wish to select

Looking For Search by

Select

User Name	Description	Forename	Surname	Active	Role Manager	Te
				Yes	No	
				Yes	No	
				Yes	No	01

Defining an Action

In the **Action** panel, define all actions selected in the **Recipient List** panel. All alerts require at least one action.

You can preview SQL alerts by selecting an action from the **Preview Action** drop down and clicking the **Preview Message** button.

6. Action

To substitute a field into either an email message, popup message, workflow message or SOAP web service, use the syntax '@OLD_Field Name' or '@NEW_Field Name' (including the quotation marks) and the value in that field will be substituted into the message. The old value of the field replaces 'OLD_Field Name' and the new value of the field replace 'NEW_Field Name'. e.g. The information for '@OLD_Surname' has been updated to '@NEW_Surname'.
NOTE: A field cannot be substituted into the message for a SQL Alert.
There is a maximum character limit: 25,000 (e-mail & workflow) and 2,000 (popup), this is for text which include SQL results.

Preview Action

Email

To define an **e-mail** action:

1. Enter a **Subject**.
2. Enter a **Message**.
3. Click the **Save** button.

e-mail

Subject

Message

Popup Message

To define a **Popup Message** action:

1. Enter a **Message**.
2. To restrict the message to users who have the **Popup** action selected in the **Recipient List**, deselect the **Show to all users** check box.
3. Click the **Save** button.

Popup Message

Message

Duration

Show to all users

Workflow Message

Before a workflow message alert can be received, a system administrator must enter the details in the **Workflow Server** tab of the Capita Children's Service Enterprise Server Configuration Utility. For more information, refer to the *Installing and Configuring One v4 Core Components* technical guide available on the One Publications website (www.onepublications.com).

To define a **Workflow Message** action:

1. Enter a **Workflow name**.
2. Enter a **Workflow message**.
3. Select a **Duration**.
4. Click the **Save** button.

Workflow Message

Workflow name

Workflow message

Duration

Portal Message

To define a **Portal Message**:

1. Select a **Template** form the drop-down.
2. Click the **Save** button.

Portal Message

Template



Administration

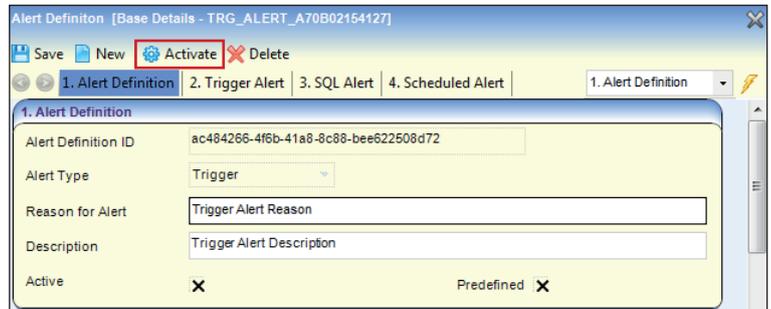
Creating Trigger Alerts

Activating an Alert

A trigger alert can be activated when it is created by entering all mandatory information, selecting the required recipients and defining at least one action. Saving the alert displays the following message: *Do you want to activate the alert?* Selecting **Yes** activates the alert immediately and the greyed out **Activate** button at the top of the page changes to **Deactivate**.

To activate an existing trigger alert:

1. Select **Tools | Administration | Alert Definition**, then select a deactivated trigger alert from the **Search** list to display the **Alert Definition** page.
2. Click the **Activate** button; the button status changes to **Deactivate**. The following message displays at the bottom of the page: *Alert activated successfully*.
You do not need to save the alert again.

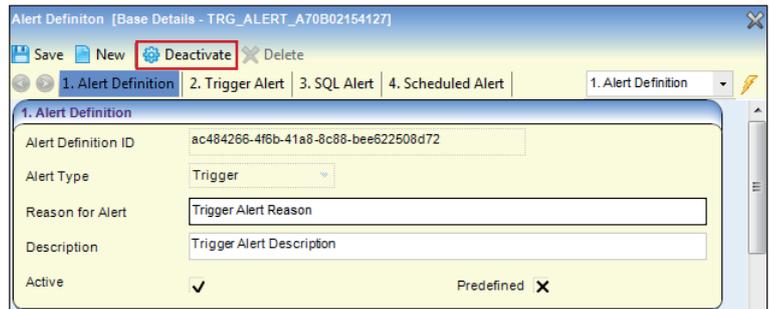


Deactivating an Alert

Predefined alerts cannot be deactivated. An existing trigger alert can be deactivated using the **Deactivate** button at the top of the page. A trigger alert cannot be changed or deleted until it has been deactivated.

To deactivate a trigger alert:

1. Select **Tools | Administration | Alert Definition**, then select a trigger alert from the browse list to display the **Alert Definition** page.
2. Click the **Deactivate** button; the button status changes to **Activate**. The following message displays at the bottom of the page: *Alert deactivated successfully*.
3. Click the **Save** button.



Deleting an Alert

Predefined alerts cannot be deleted. A trigger alert cannot be deleted until it has been deactivated.

To delete a scheduled alert:

1. Select **Tools | Administration | Alert Definition**, then select a trigger alert from the browse list to display the **Alert Definition** page.
2. If required, click the **Deactivate** button; the button status changes to **Activate**. The following message is displayed at the bottom of the page: *Alert deactivated successfully*.
3. Click the **Delete** button; the following message displays: *Are you sure you want to delete this <Alert> for the <Table>?*
4. Click the **Yes** button; the following message is displayed at the bottom of the page: *Alert <Name> for Table <Name> is deleted successfully*.



Related Reference Guides:

- [RG_Administration_Creating SQL Alerts](#)
- [RG_Administration_Creating Scheduled Alerts](#)
- [RG_Administration_System Alerts](#)