



CNS Audit

Version 1.0

Setup Guide

Table of Contents

Preparing the database for CNS Audit	3
Set up the database for tracking field changes	6
Set up the database for tracking record deletions	11
Set up the database for startup and shutdown	14
Optional set up (1)	20
Optional set up (2)	20
Set up is complete	23
Additional Information	23
Modifying schema	23
Protecting Audit Trail Data	23
Auditing Actions other than Modifying Data	24
Auditing Record Views	24
Preventing database modification if CNS Audit is not present	26

Preparing the database for CNS Audit

In order for CNS Audit to work properly, there cannot be any Table Occurrences with the same name as Layouts in the database. CNS Audit is capable of keeping track of the fields in the database even if their names are changed, but to do this it needs to use the FieldIDs calculation function. This function allows for retrieving the Field IDs of all the fields in a specific Table Occurrence, so long as there is not a Layout with the same name as the Table Occurrence. If there is a Layout with the same name as the Table Occurrence, the FieldIDs function returns the Field IDs of all the fields on the Layout instead of in the Table Occurrence. Since all the fields in the Table Occurrence may not be on the Layout, this causes CNS Audit to not see, and therefore not Audit, all the fields in the Table Occurrence.

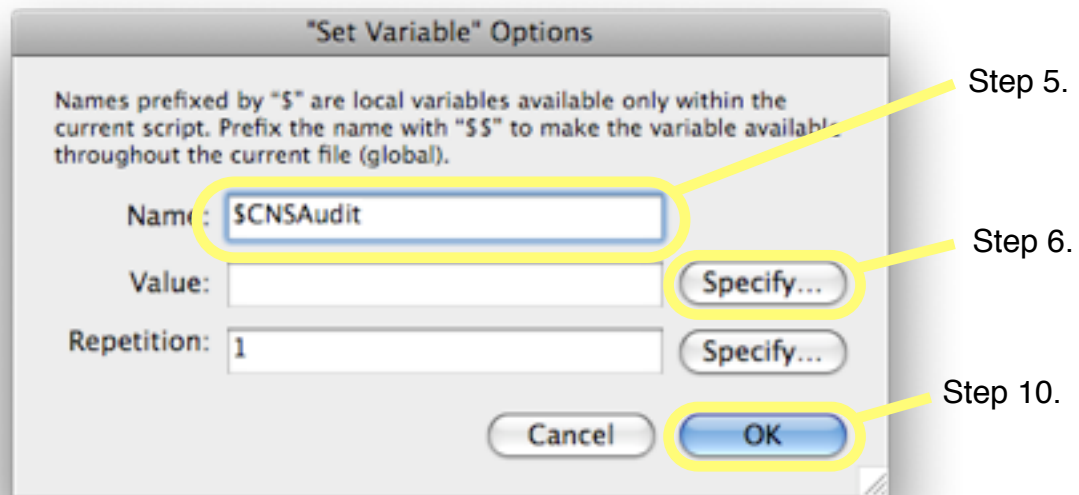
Unfortunately, when creating a table, FileMaker Pro automatically creates a Layout with the same name as the table. To prepare a database for use with CNS Audit, these duplicate names must be changed. Either rename the Table Occurrence or rename the Layouts. It is probably easier to rename the Layouts because when renaming a Table Occurrence, if the automatically generated Layout has never been manually renamed, FileMaker Pro updates the automatically generated Layout to have the same name as the Table Occurrence. Consider renaming the Layout by adding the word "Entry" to the end. For example, if a Table Occurrence is named *Contacts*, the associated Layout would be named *Contacts Entry*. However, this is only a suggestion.

To determine which Layouts have the same name as Table Occurrences in the database, either make a list of all the Table Occurrences in the database and then check for duplicate Layout names, or let CNS Audit create a list for you:

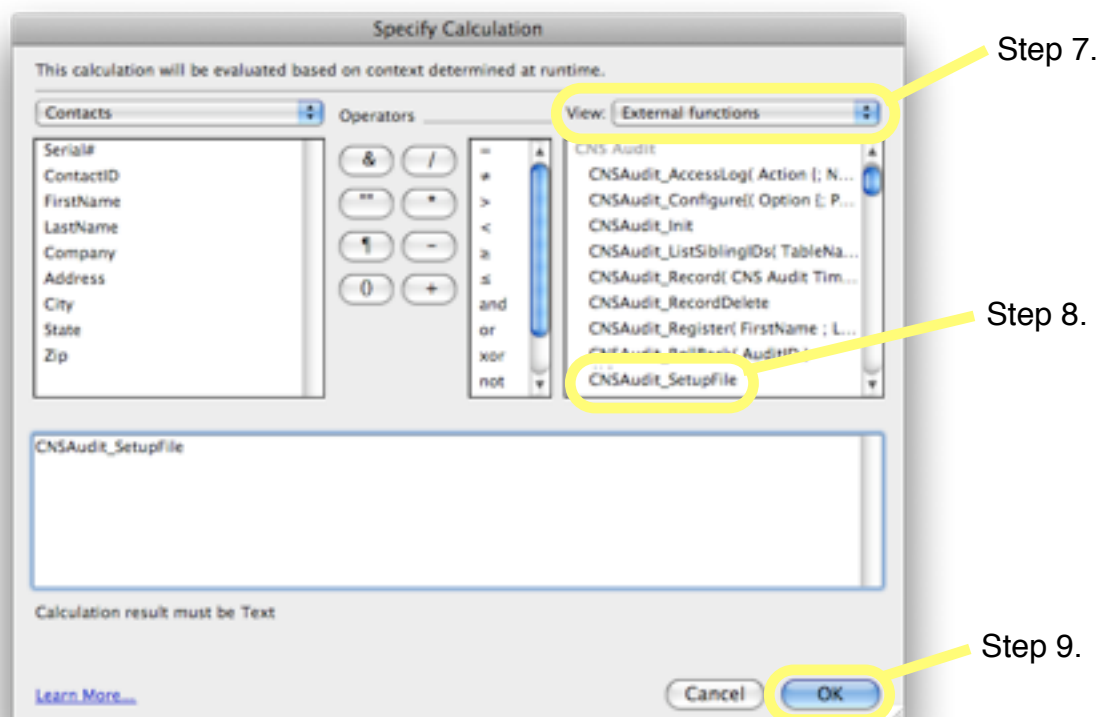
1. Open the database, and log in with a [Full Access] user account.
2. Open the ScriptMaker / Manage Scripts dialog.
3. Create a new script named Setup Auditing Preferences or something similar.
4. Add a Set Variable script step and double-click it.¹

¹ If using FileMaker Pro 7, use a Set Field script step and set a field in the database with the same calculation as the Value calculation for the Set Variable script step.

5. In the Set Variable Options dialog, name the variable \$CNSAudit or something similar.

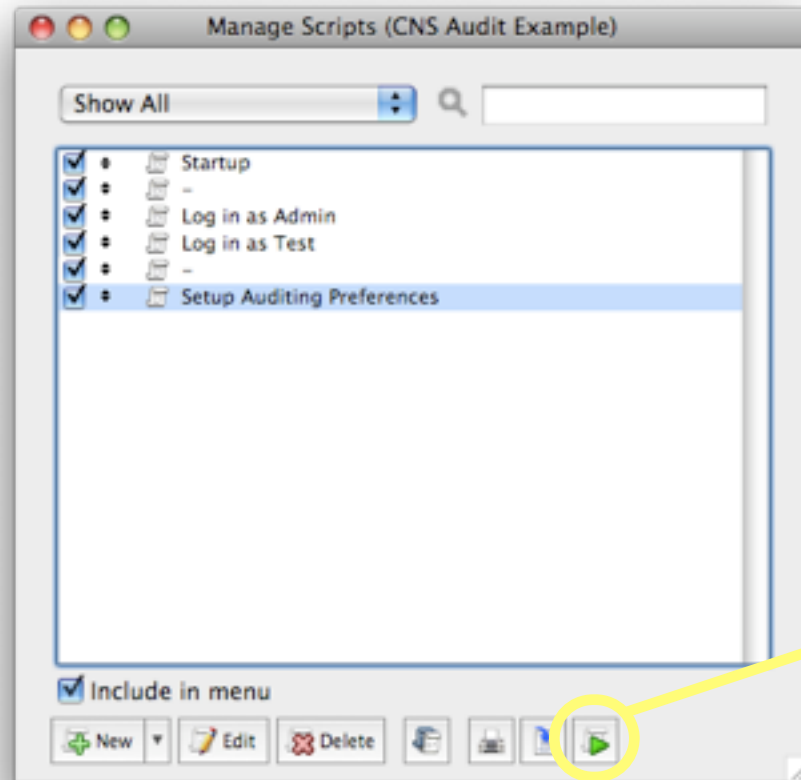


6. Press Specify... next to the Value field.
7. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.



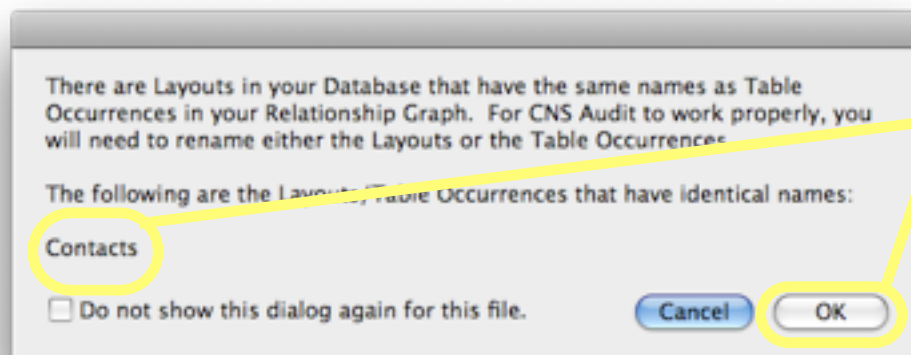
8. In the list of functions, scroll down until you see a line for CNSAudit_SetupFile and double-click it.
9. Press OK to close the Specify Calculation dialog.

10. Press OK to close the Set Variable Options dialog.
11. Save and close the Setup Auditing Preferences script.
12. Select the Setup Auditing Preferences script (if it's not already selected), and press Perform.



Step 12.

13. If there are any Layouts with the same name as Table Occurrences, CNS Audit should present a dialog warning you about it and list all the conflicting names.
14. Make a list of the conflicting names and press OK.



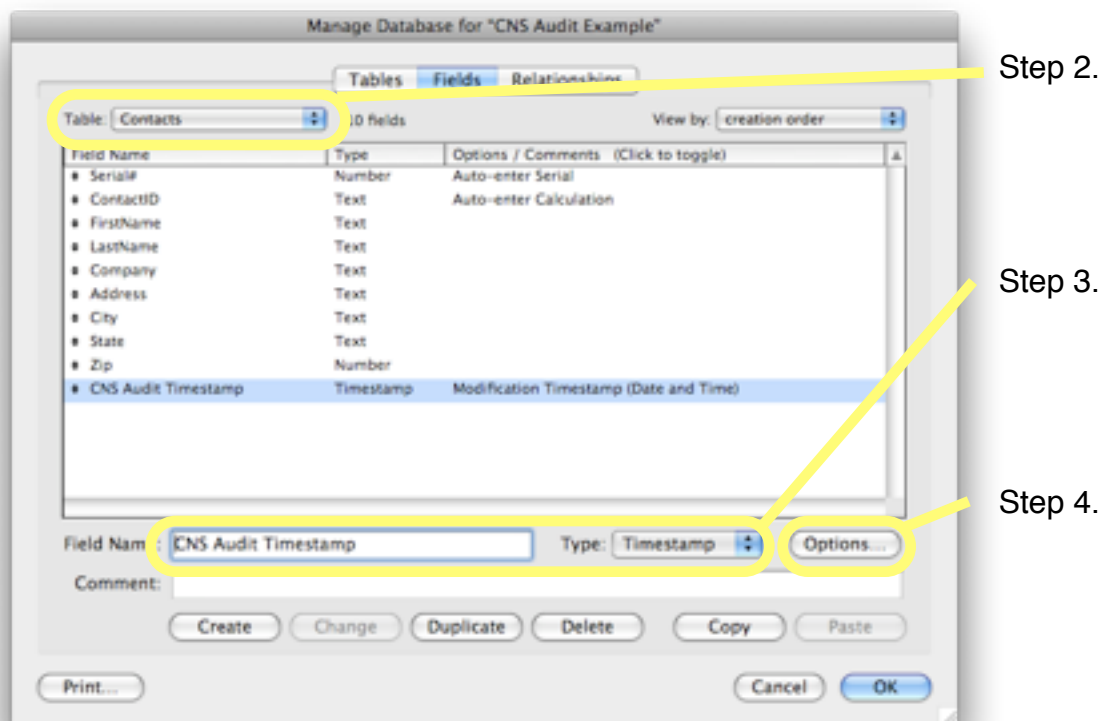
Step 14.

15. Press Close to close the File Setup dialog.
16. Rename the conflicting Table Occurrences and/or Layouts.
17. Repeat steps 12-16 to ensure all conflicts have been resolved.

Set up the database for tracking field changes

This section explains the steps necessary to set up each Table in the database for auditing. After completing these steps, the plug-in will be able to track the changes to field values in the database. (Note: You may also want to implement the validation options described in the *Preventing database modification if CNS Audit is not present* subsection of the *Additional Information* section at this time.)

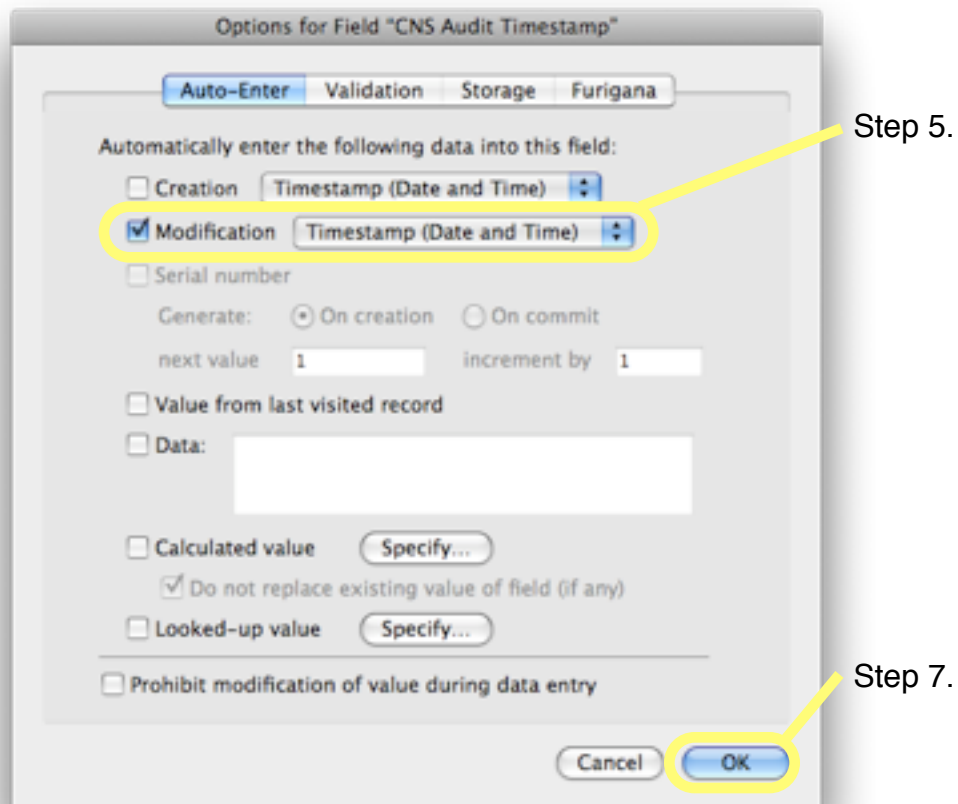
1. Open the Database file and go to Define / Manage Database.
2. Switch to the Fields tab and select the first table from the Table drop-down menu in which auditing needs to be added.



3. Add a new Timestamp field named CNS Audit Timestamp.²
4. Press Options for the CNS Audit Timestamp field.

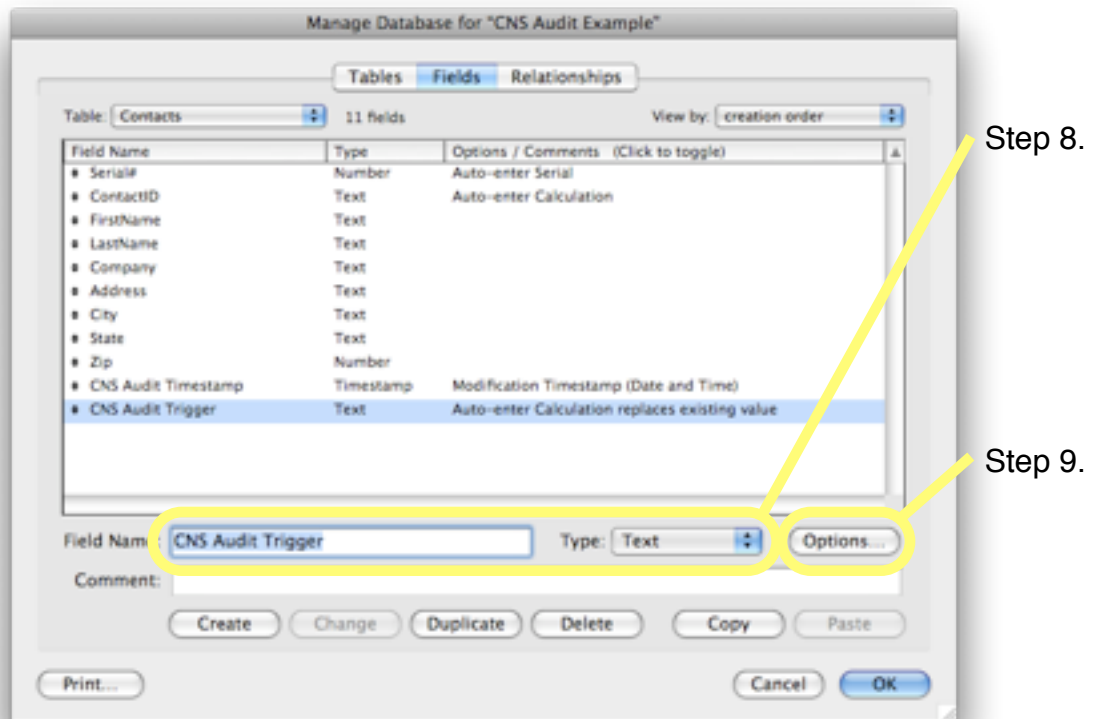
² Make sure the field is named CNS Audit Timestamp exactly.

5. In the Options for Field dialog that pops-up, switch to the Auto-Enter tab, and check the Modification Timestamp option.



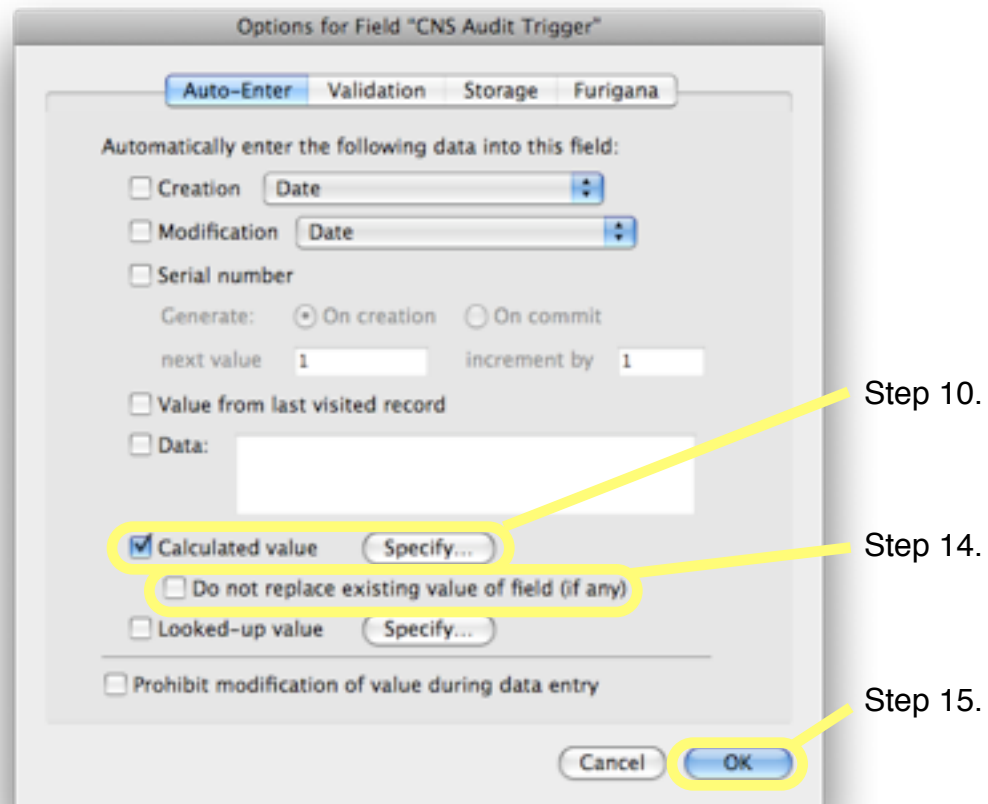
6. *Optional:* Set up the Validation options as described in the *Preventing database modification if CNS Audit is not present* subsection of the *Additional Information* section.
7. Press OK to close the Options for Field dialog.
8. Add a new Text field named CNS Audit Trigger.³

³ Make sure the field is named CNS Audit Trigger exactly.

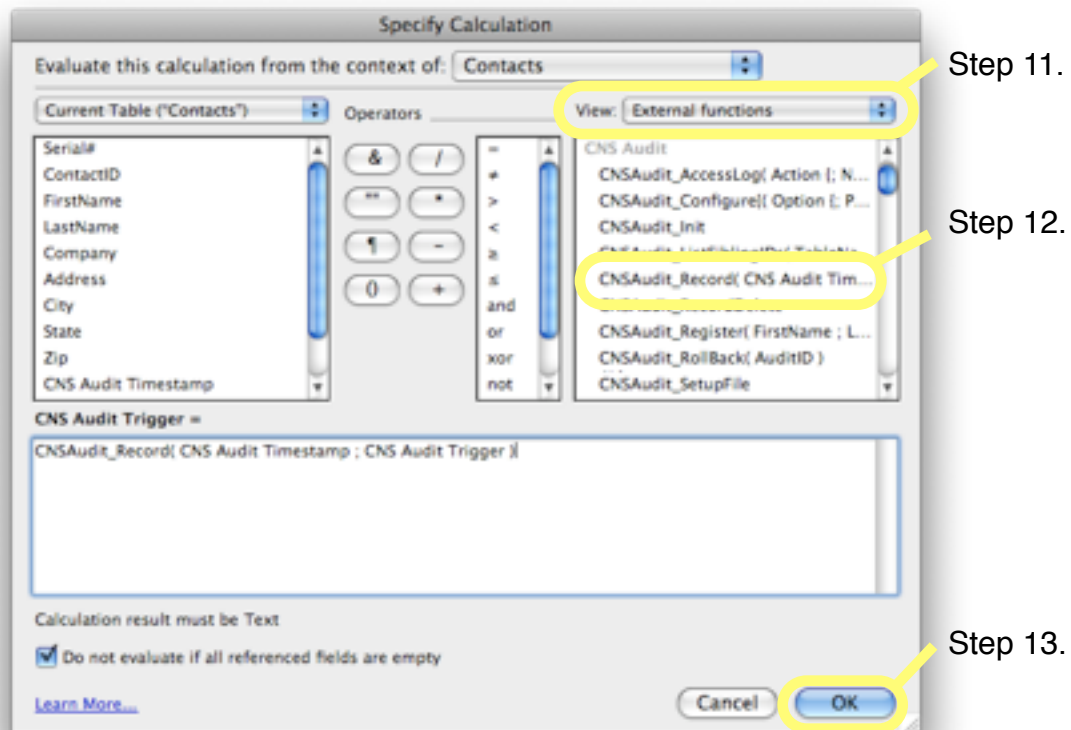


9. Press Options for the CNS Audit Trigger field.

10. In the Options for Field dialog that pops-up, switch to the Auto-Enter tab, and check the Calculated Value option.



11. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.



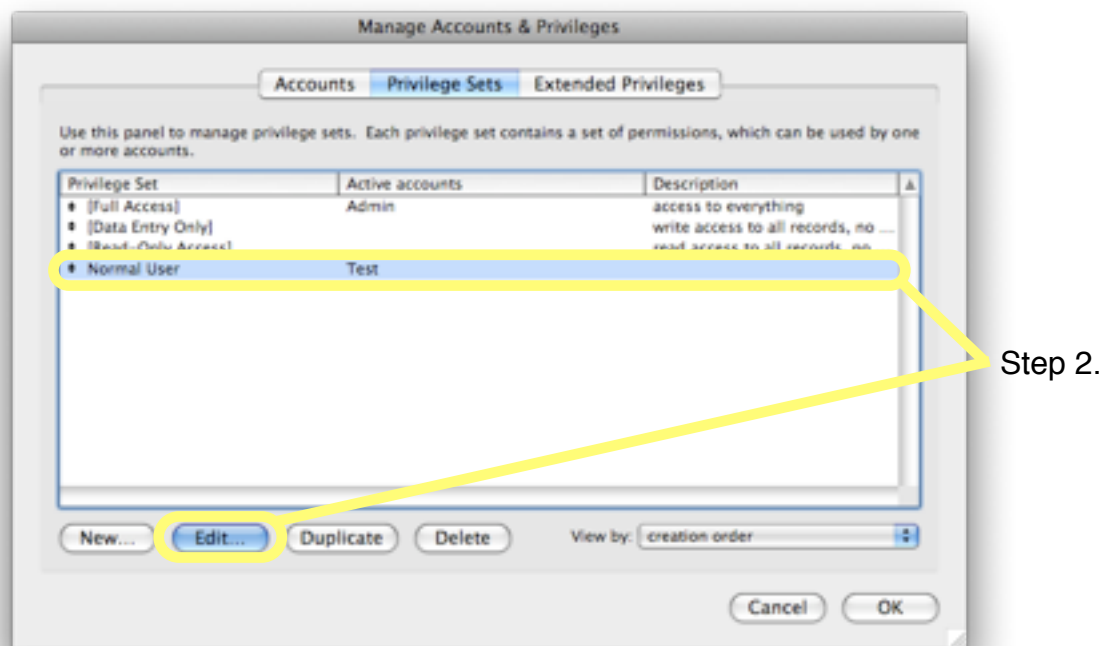
12. In the list of functions, scroll down until a line for CNSAudit_Record is visible and double-click it.
13. Press OK to close the Specify Calculation dialog.
14. Back on the Options for Field dialog, uncheck the Do not replace existing value of field (if any) option.
15. Press OK to close the Options for Field dialog.
16. Repeat steps 3-15 for each table that needs auditing. (If using FileMaker Pro Advanced, highlight the two fields, press Copy, and then go to each table and press Paste.)
17. Close the Define / Manage Database dialog.

Set up the database for tracking record deletions

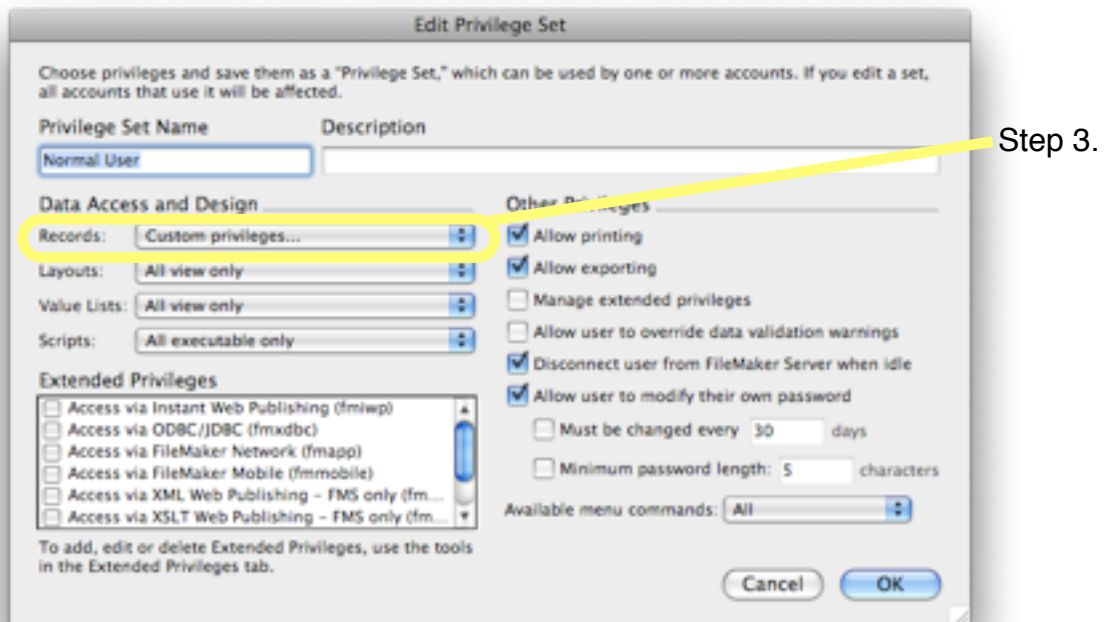
This section explains the steps necessary to set up each Privilege Set for tracking record deletions.

Note: The Record Privileges for the [Full Access] Privilege Set cannot be modified, therefore any user logging in with the [Full Access] Privilege Set will be able to delete records without being audited. Consider duplicating the [Full Access] Privilege Set and adding the necessary options below to track record deletions, then assign the Privilege Set to the users who need full access.

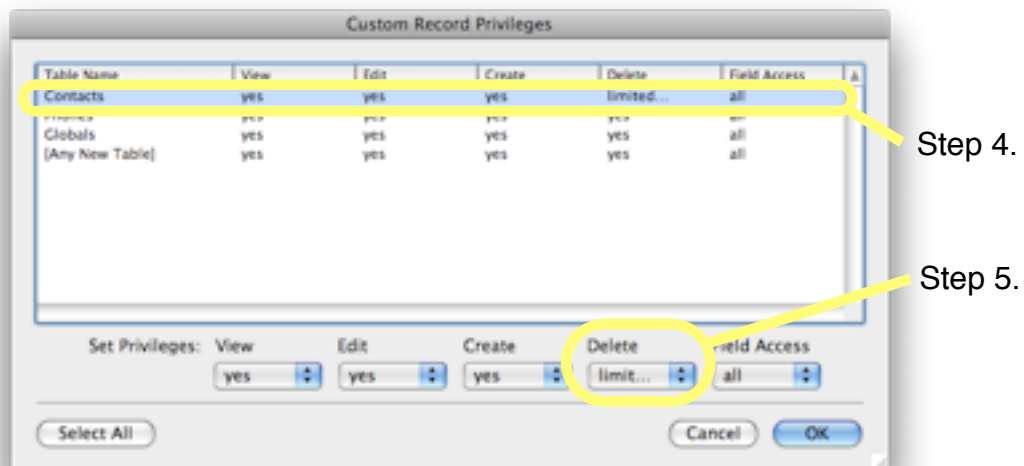
1. Open up the Accounts & Privileges dialog.
2. Switch to the Privilege Sets tab and edit the first custom privilege set. If there are no custom privilege sets, highlight the [Data Entry Only] privilege set, press Duplicate, and then edit that new privilege set.



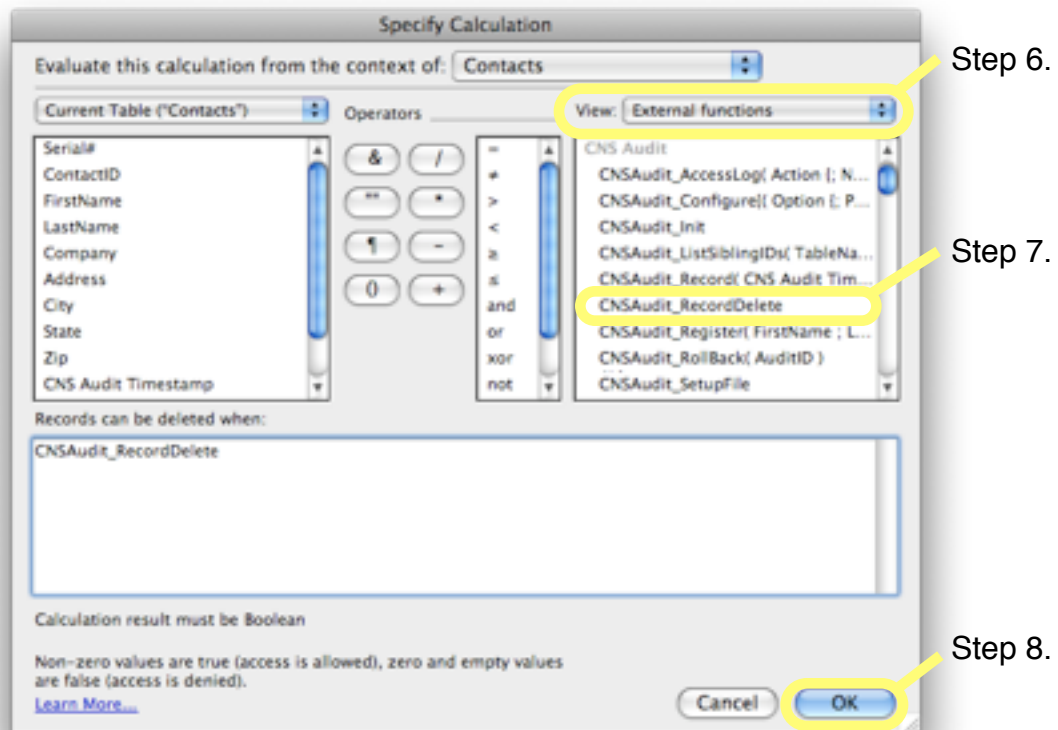
3. In the Data Access and Design section, press the Records drop-down menu and select Custom Privileges.



4. In the Custom Record Privileges dialog that pops up, select the first table.
5. Press the Delete drop-down menu and select limited....



6. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.

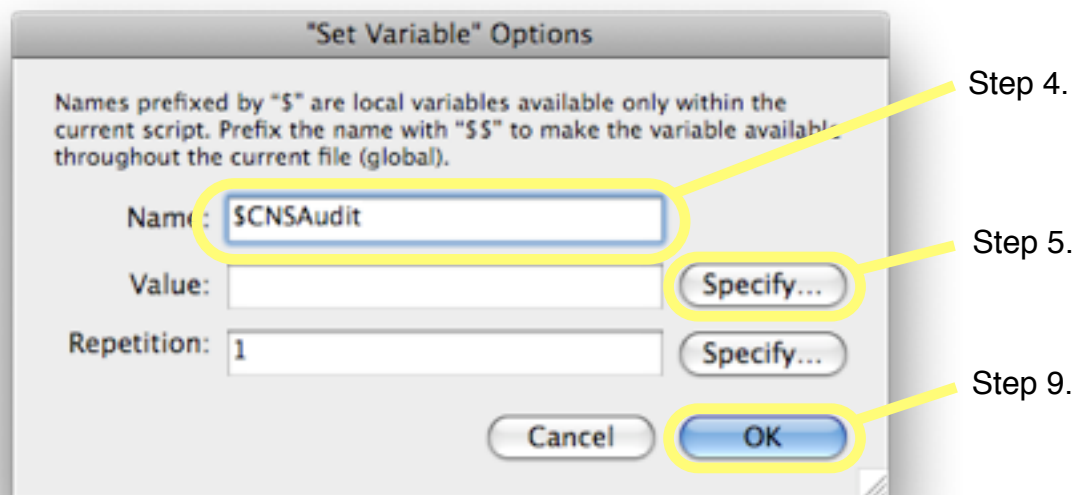


7. In the list of functions, scroll down until a line for CNSAudit_RecordDelete is visible and double-click it.
8. Press OK to close the Specify Calculation dialog.
9. Repeat steps 5-8 for each table in the Custom Record Privileges dialog. (Once the CNSAudit_RecordDelete function has been added to one calculation, copy and paste it to the other ones, or just retype it, and skip selecting the External Functions repeatedly.)
10. Press OK to close the Custom Record Privileges dialog.
11. Press OK to close the Edit Privilege Set dialog.
12. Repeat steps 3-12 for each custom privilege that needs modifying.
13. Switch to the Accounts tab and reassign Privilege Sets to the Accounts as necessary.
14. Press OK to close the Accounts & Privileges dialog.

Set up the database for startup and shutdown

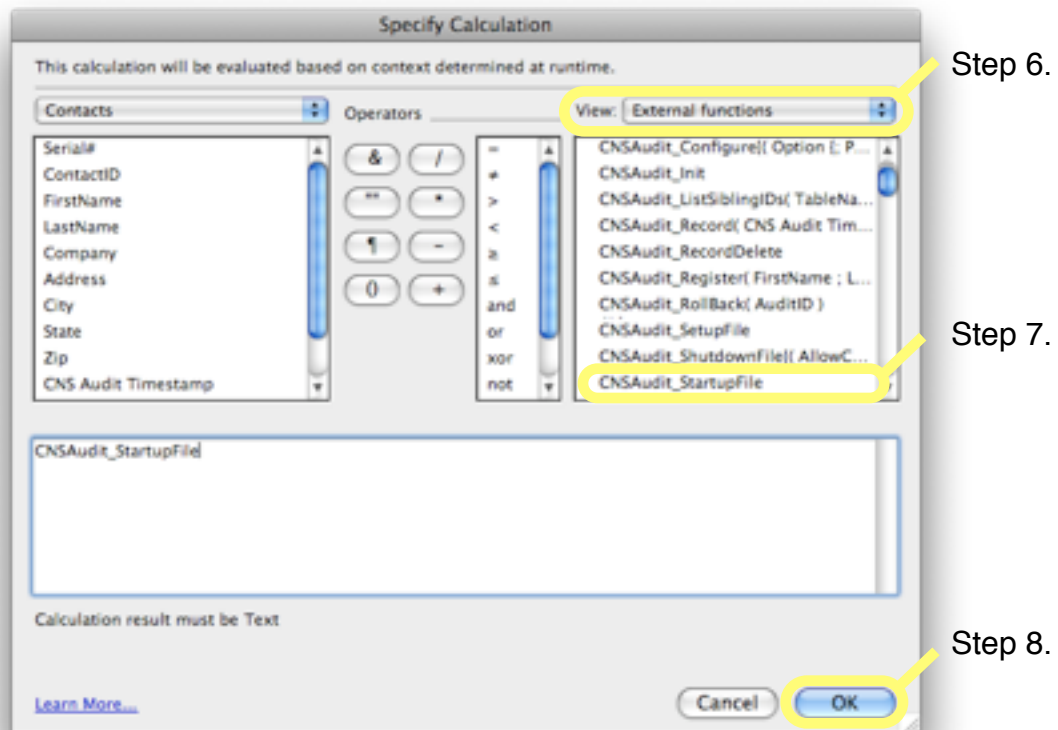
This section explains how to add or modify the Startup and Shutdown scripts for the database for setting up CNS Audit for the file. While this is optional, by following these instructions, CNS Audit will function better with the database.

1. Open up the ScriptMaker / Manage Scripts dialog.
2. Create a script named Startup (or something similar) or edit an existing startup script if one is already in use.
3. Add a Set Variable script step and double-click it.⁴
4. In the Set Variable Options dialog that pops up, name the variable \$CNSAudit or something similar.



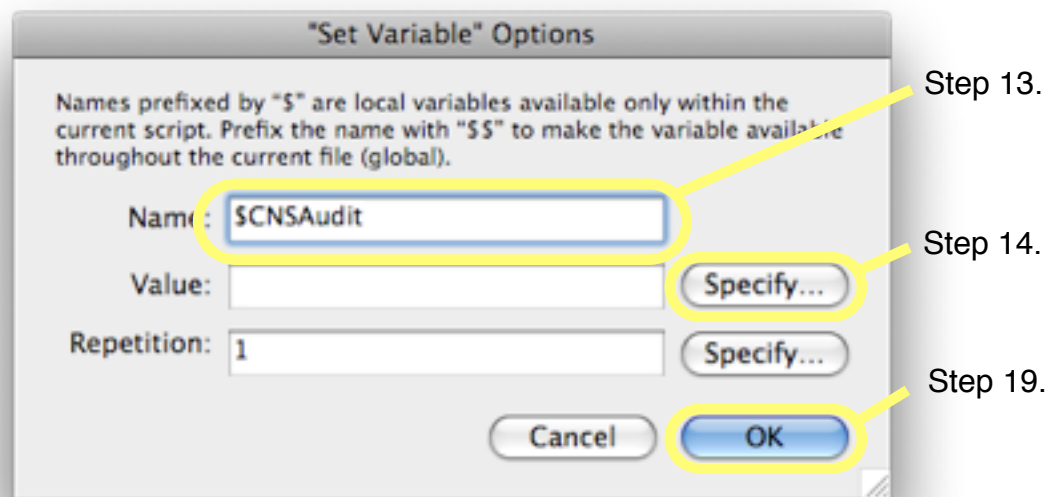
5. Press Specify... next to the Value field.
6. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.

⁴ If using FileMaker Pro 7, use a Set Field script step and set a field in the database with the same calculation as the Value calculation for the Set Variable script step.



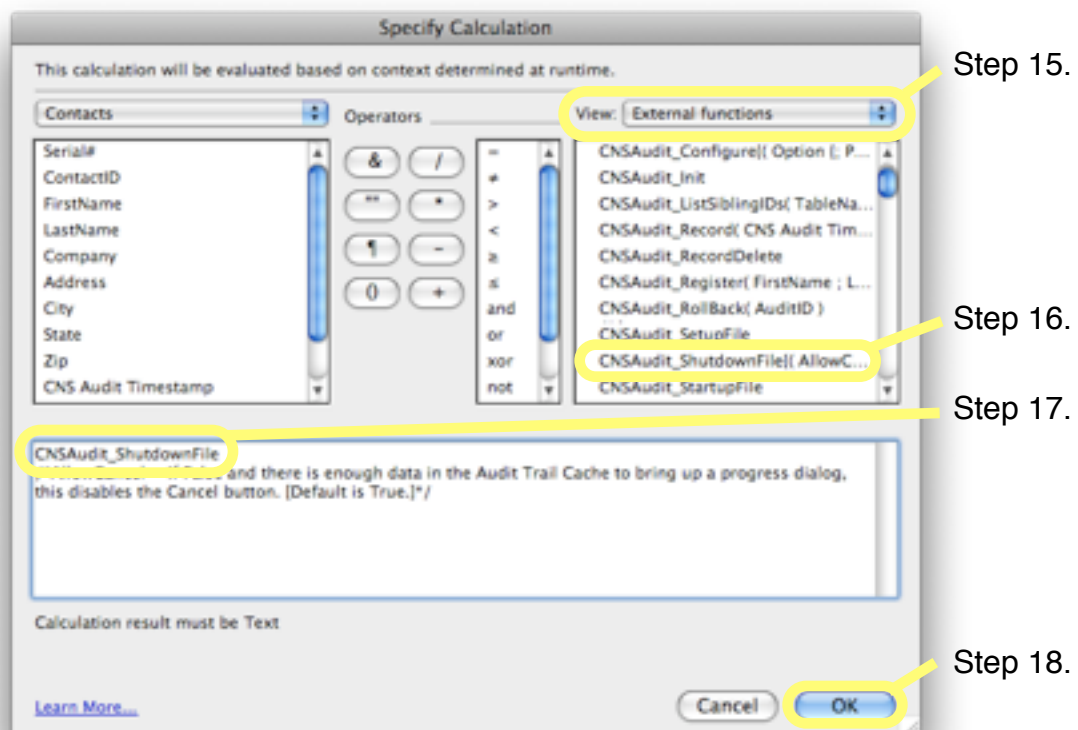
7. In the list of functions, scroll down until a line for CNSAudit_StartupFile is visible and double-click it.
8. Press OK to close the Specify Calculation dialog.
9. Press OK to close the Set Variable Options dialog.
10. Save and close the Startup script window.
11. Create a script named Shutdown (or something similar) or edit an existing shutdown script if one is already in use.
12. Add a Set Variable script step and double-click it.⁴

13. In the Set Variable Options dialog that pops up, name the variable \$CNSAudit or something similar.



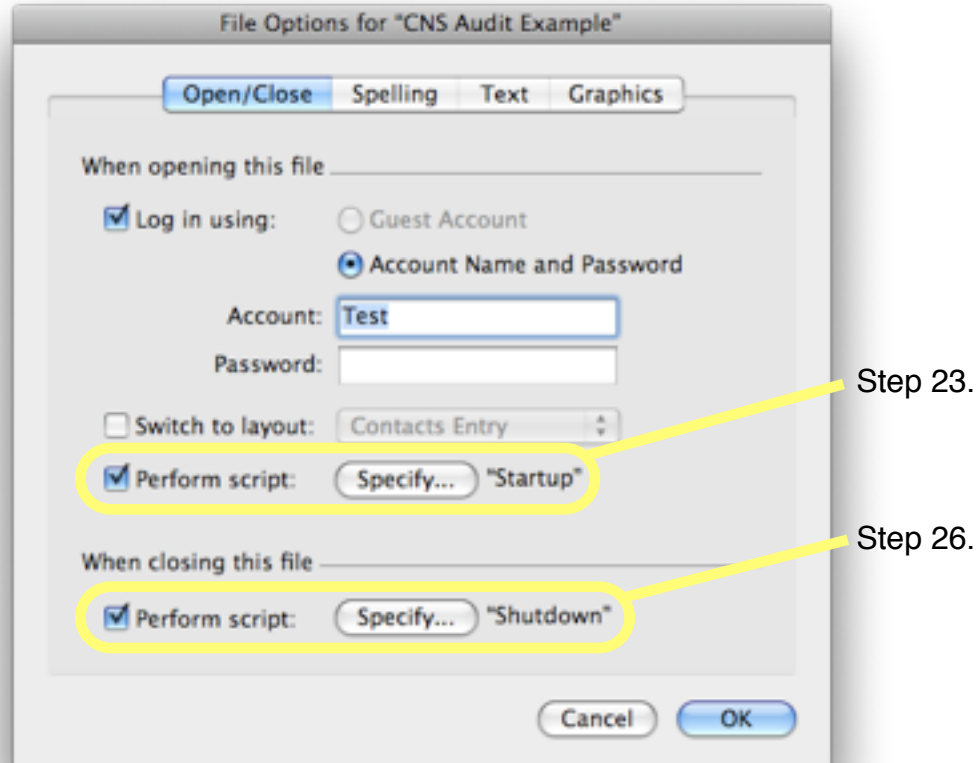
14. Press Specify... next to the Value field.

15. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.



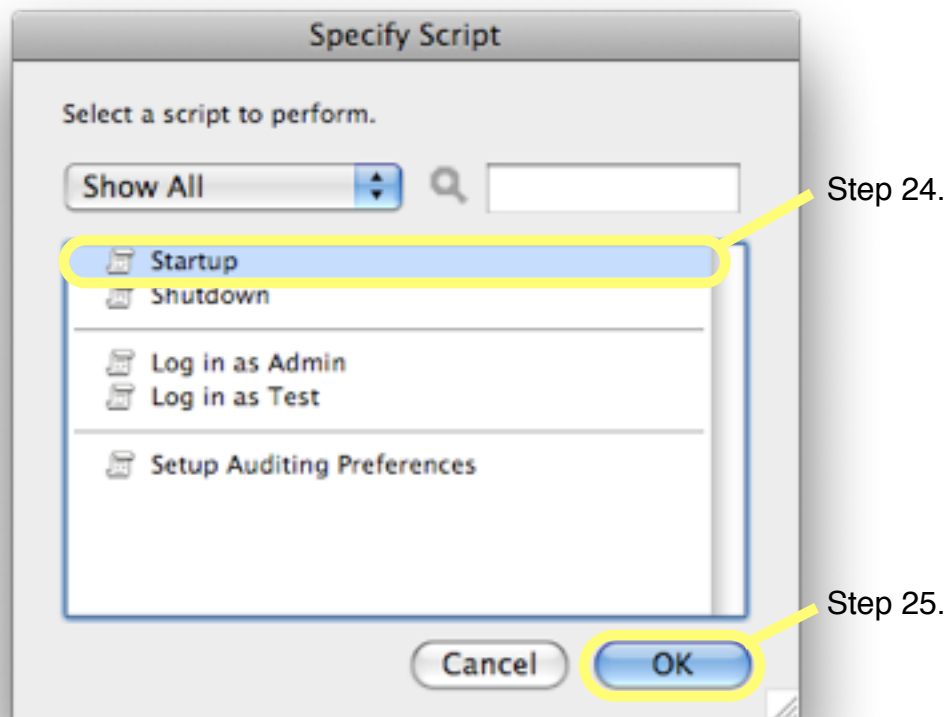
16. In the list of functions, scroll down until a line for CNSAudit_ShutdownFile is visible and double-click it.

17. For now, select everything in curly brackets after CNSAudit_ShutdownFile and delete it.⁵
18. Press OK to close the Specify Calculation dialog.
19. Press OK to close the Set Variable Options dialog.
20. Save and close the Shutdown script window.
21. Close the ScriptMaker / Manage Scripts dialog.
22. Open the File Options dialog and switch to the Open/Close tab.
23. In the When opening this file section, check the Perform script option.



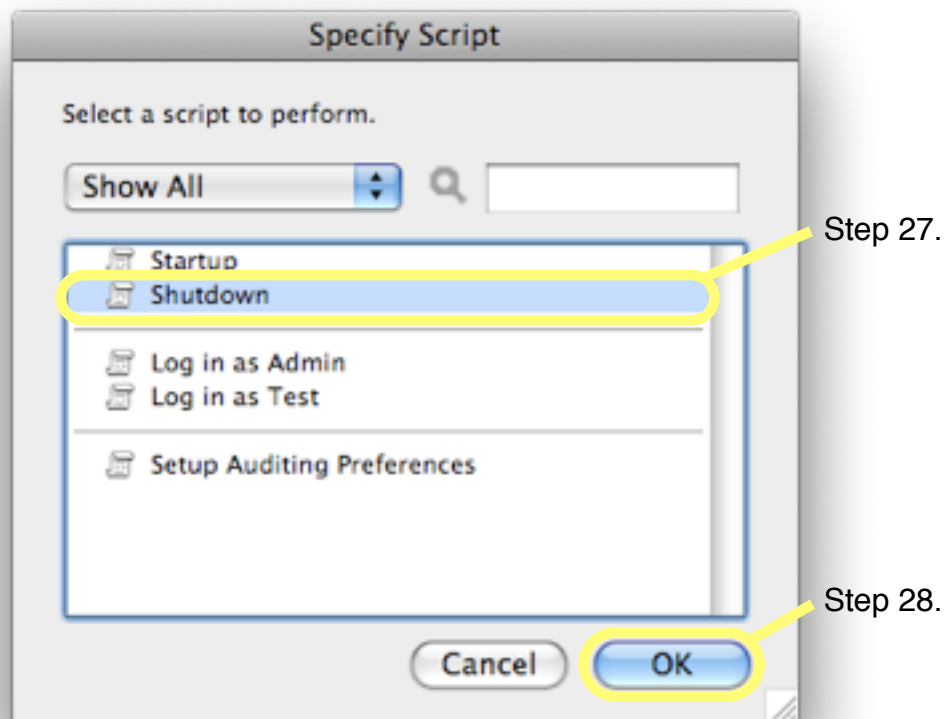
⁵ See the online CNS Audit Function Reference for an explanation of the optional parameter for the CNSAudit_ShutdownFile function.

24. In the Specify Script dialog that pops up, select the Startup script created/edited in step 2.



25. Press OK to close the Specify Script dialog.
26. Back on the File Options dialog, in the When closing this file section, check the Perform script option.

27. In the Specify Script dialog that pops up, select the Shutdown script created/edited in step 11.



28. Press OK to close the Specify Script dialog.

29. Press OK to close the File Options dialog.

Optional set up (1)

Consider closing and reopening the database to ensure all the above changes are committed to the database file.

Optional set up (2)

CNS Audit is invoked when a record is committed and at that point it compares the data in the committed record with the Audit Trail data in the CNS Audit Log database. When adding CNS Audit to an existing database with existing data (as opposed to a new database), the CNS Audit Log will not contain any Audit Trail data for the database at the point CNS Audit is implemented. If there are no previous entries for any given record in the CNS Audit Log when a record is audited, CNS Audit will create new entries for the record, but any values in the record prior to the audit will be lost.

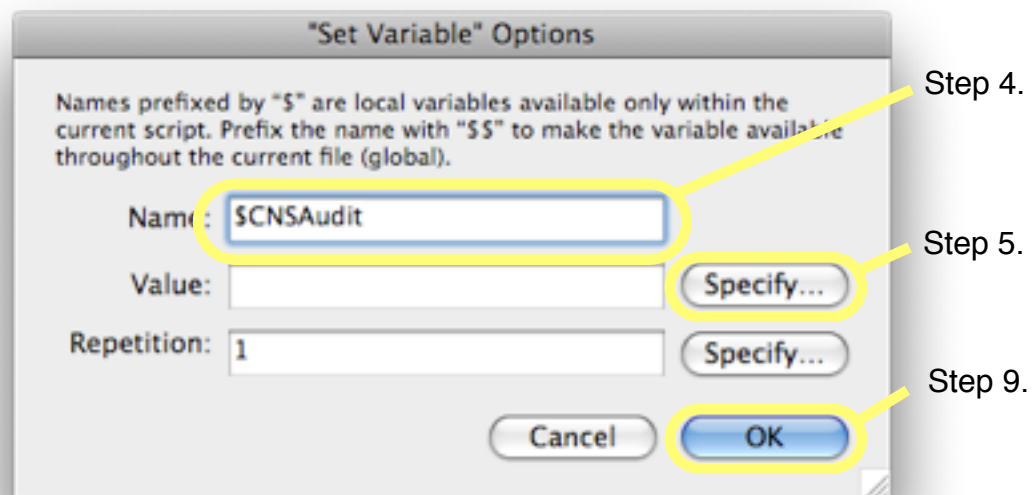
To remedy this situation, CNS Audit includes a CNSAudit_Init function for creating an initial set of Audit Trail data. This function will look for every table in your database that contains the CNS Audit Trigger field (ie. tables that have been set up for auditing) and examine every record in those tables to determine if that record has been audited. If it has not, it creates a set of initial records in the CNS Audit Log table to reflect the state of the record at that moment. After using CNSAudit_Init the CNS Audit Log table will contain all necessary information for rolling back changes to the database up to the point that CNS Audit was implemented.

Follow these steps to use the CNSAudit_Init function:

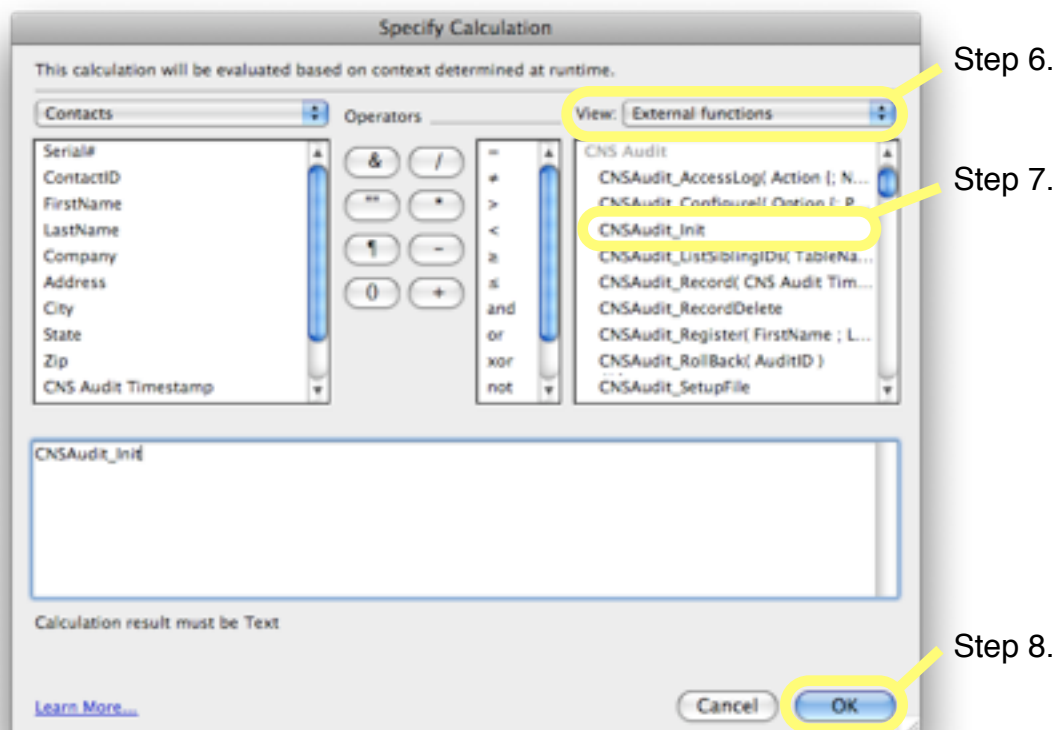
1. Open up the ScriptMaker / Manage Scripts dialog.
2. Create a script named Init File or something similar.
3. Add a Set Variable script step and double-click it.⁶

⁶ If using FileMaker Pro 7, use a Set Field script step and set a field in the database with the same calculation as the Value calculation for the Set Variable script step.

4. In the Set Variable Options dialog that pops up, name the variable \$CNSAudit or something similar.



5. Press Specify... next to the Value field.
6. In the Specify Calculation dialog that pops up, change the View drop-down to External Functions.



7. In the list of functions, scroll down until a line for CNSAudit_Init is visible and double-click it.
8. Press OK to close the Specify Calculation dialog.

9. Press OK to close the Set Variable Options dialog.
10. Save and close the Init File script window.

Run the Init File script to initialize the CNS Audit Log table with the data from the database. If the CNS Audit Log table has not yet been created, this function will create it and ask you to run the Init function a second time.

Notes:

- Depending on the size of the existing database, the CNSAudit_Init function can take a long time to complete. While running, CNSAudit_Init shows a dialog with a progress bar and a Cancel button. The initialization process can be stopped at any time by pressing the Cancel button. The process can then be started again by running CNSAudit_Init again and the function will pick up where it left off.
- Audit Trail data can be very large. The CNS Audit Log table contains a record for every field change that occurs along with several fields containing state information about those field changes. Using the CNSAudit_Init function will create records in the CNS Audit Log for every field in every record in every table in the database. Consider only using this function if Audit Trail data of this magnitude is actually needed.
- The CNSAudit_Init function can be called at any time after CNS Audit has been implemented in a database and can be called multiple times if the need arises. For example, if for some reason CNS Audit was not installed on a computer that was using the database, any records added on that computer would not be audited. The CNSAudit_Init function could be used to create Audit Trail data for those records added on the computer that did not have CNS Audit installed.

Set up is complete

Begin editing the data. CNS Audit will automatically create the CNS Audit Log table in the database, if it has not done so already, and begin adding audit trail data to it. If the CNS Audit Log table is not appearing, try logging into your database with a [Full Access] account.

Additional Information

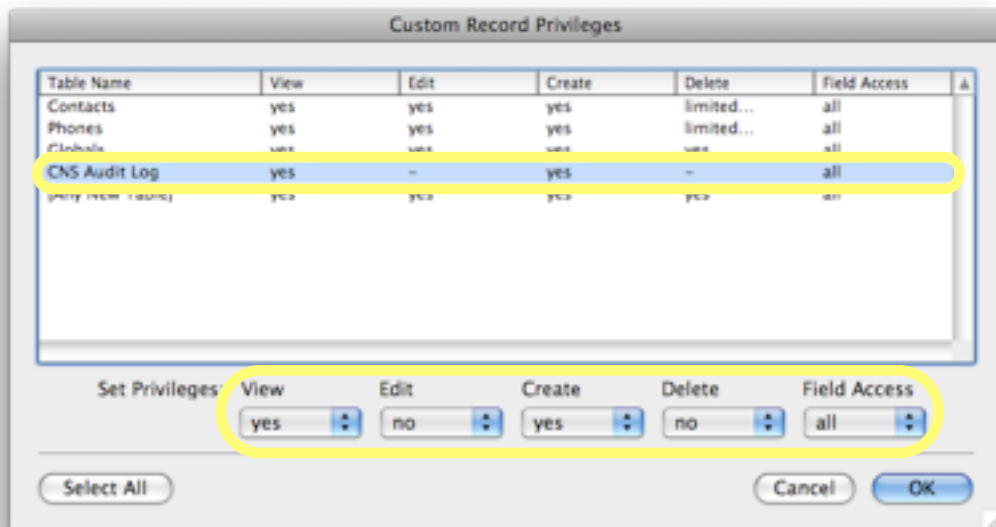
Modifying schema

One of the many steps CNS Audit takes to speed up auditing is to cache the schema information from the database. If you are actively developing the database (eg. adding/renaming tables, fields, and layouts), this cached schema information can become out of sync with the actual database. To ensure that CNS Audit is correctly auditing the database, the CNSAudit_StartupFile function can be used at any time to reload the schema information.

Protecting Audit Trail Data

While CNS Audit is in charge of creating an Audit Trail of the database, keeping everyday users from directly modifying that Audit Trail data after-the-fact is up to the database developer. To ensure that everyday users can add to, but not modify, the Audit Trail data, modify the CNS Audit Log entry in the Custom Record Privileges for the everyday users' Privilege Set as follows:

View:	yes
Edit:	no
Create:	yes
Delete:	no
Field Access:	all

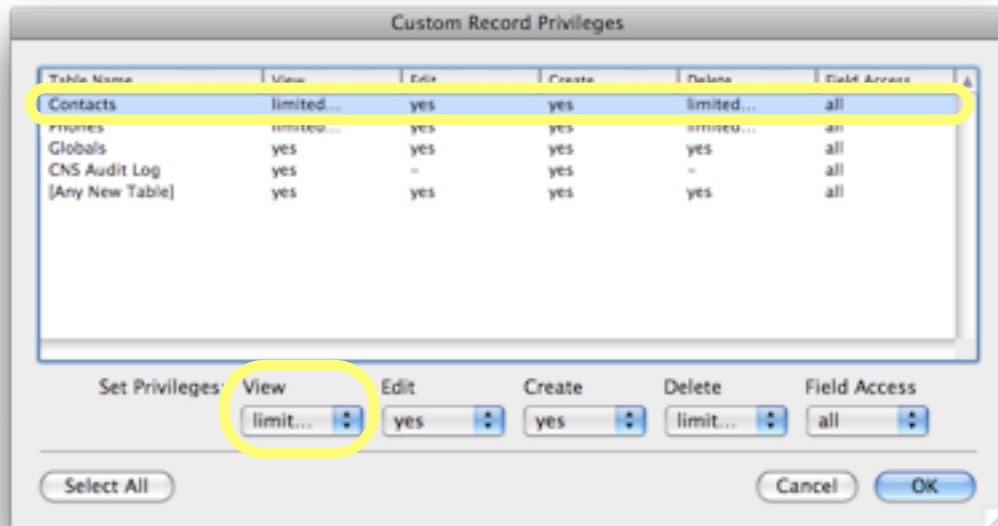


Auditing Actions other than Modifying Data

CNS Audit provides a `CNSAudit_AccessLog` function for auditing any custom actions performed by database users. This function has one required parameter and one optional parameter. The first parameter is the Action to log, for example User Login. The second parameter is any special Notes about the action, for example User is logged in after hours. This information is stored in a separate table in the database named CNS Access Log. If the CNS Access Log table is not appearing, try logging into the database with a [Full Access] account.

Auditing Record Views

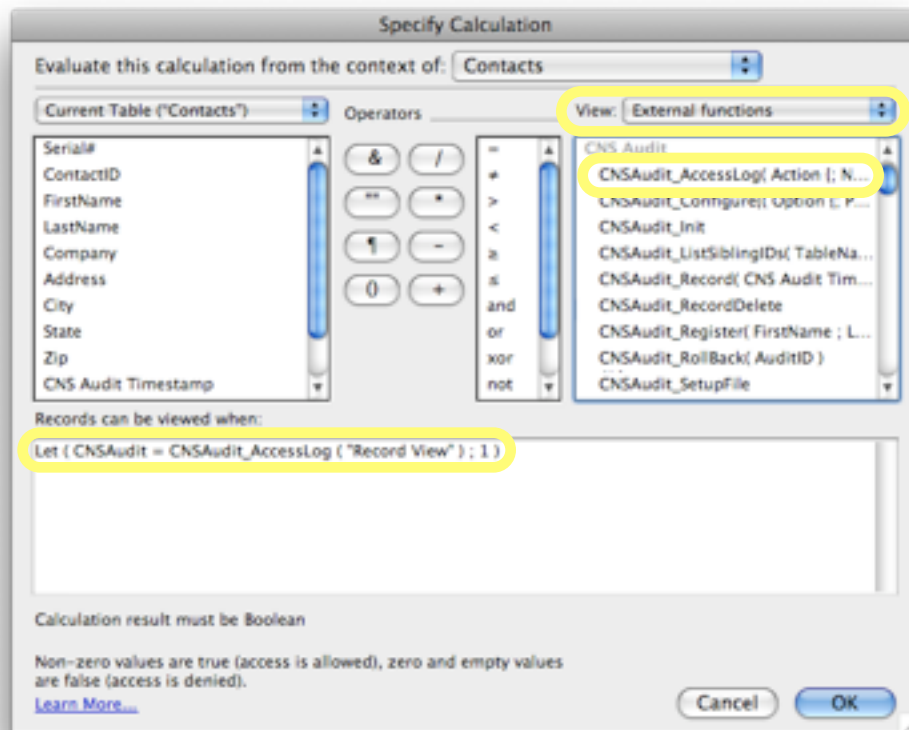
Auditing Record Views can be achieved by modifying the View privilege for any table in the Custom Record Privileges dialog for the Privilege Sets of the database users.



Select the limited... privilege and use a calculation similar to the following:

```
Let ( CNSAudit = CNSAudit_AccessLog( "Record View" ) ; 1 )
```

The Let statement allows for calling the CNSAudit_AccessLog function and discarding the result (the CNSAudit variable is not used), while the 1 allows the Record to be Viewed as normal.

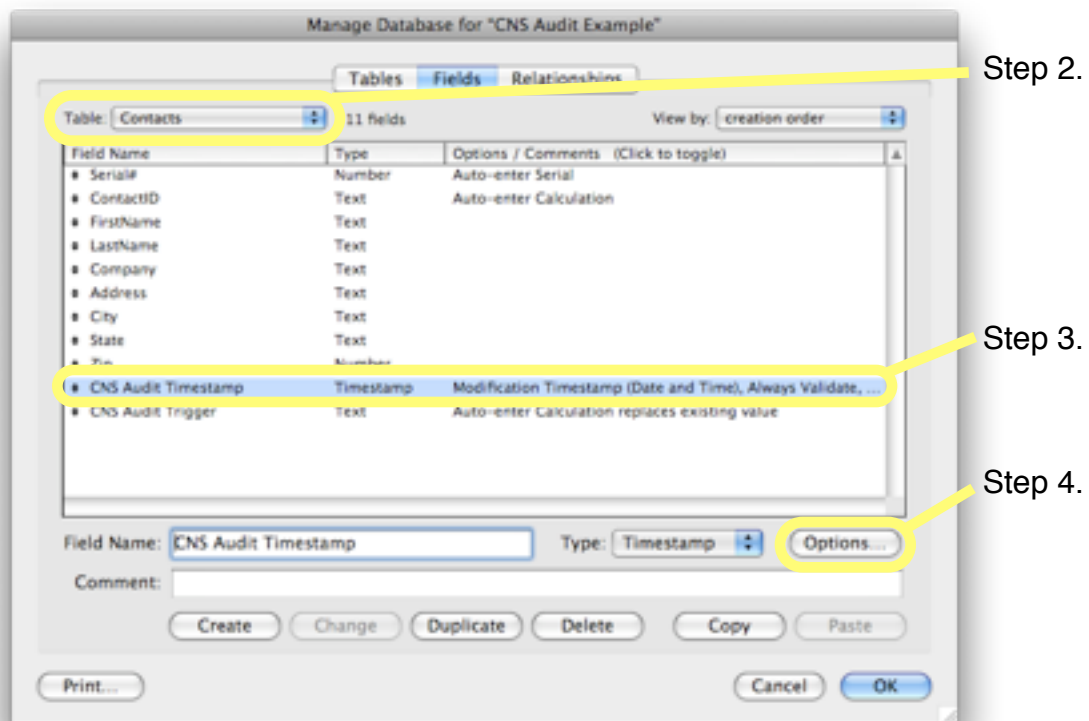


Note: The Record ID is logged by CNS Audit as one of the standard log fields, so there is no need to try and include the Record ID within the call to CNSAudit_AccessLog.

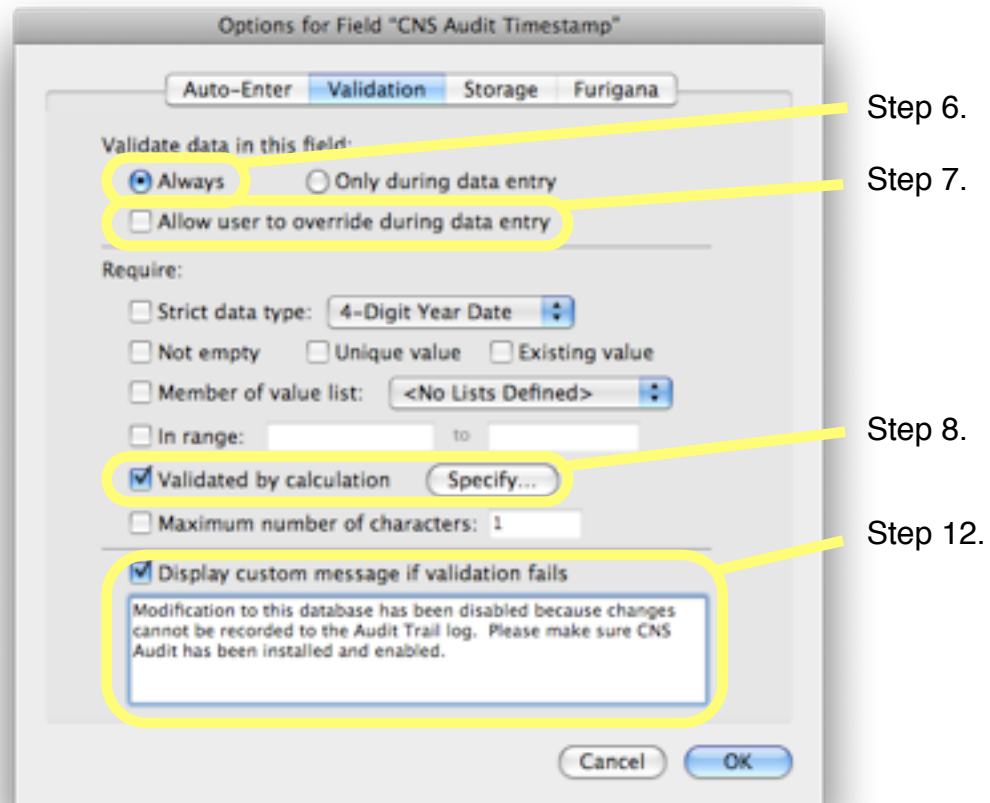
Preventing database modification if CNS Audit is not present

If CNS Audit is not installed or not enabled on a machine, any modifications to the database will not be recorded to the CNS Audit Log. To correct this behavior, some validation options can be added to prevent modification to the database. Follow these steps to prevent modification when CNS Audit is missing:

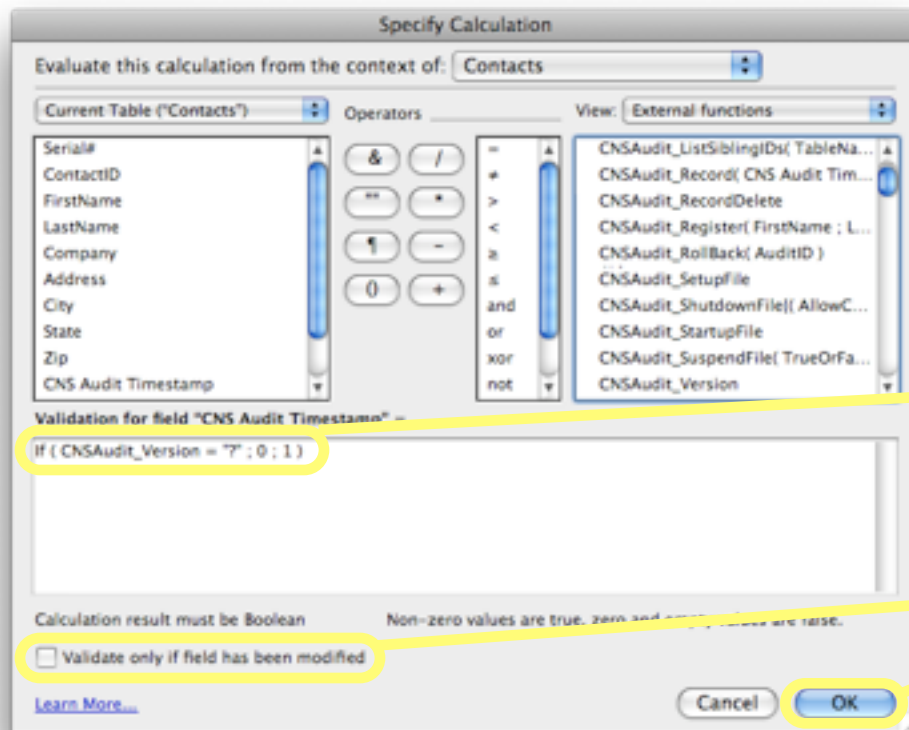
1. Open the Database file and go to Define / Manage Database.
2. Switch to the Fields tab and select the first table from the the Table drop-down menu in which auditing has been added.



3. Select the CNS Audit Timestamp field that was created when the database was set up for tracking field changes.
4. Press Options.
5. In the Options for Field dialog that pops-up, switch to the Validation tab.
6. Select Always in the Validate data in this field: section.



7. Uncheck the Allow user to override during data entry option in the Validate data in this field: section.
8. Check the Validated by calculation option in the Require: section.
9. In the Specify Calculation dialog that pops up, type into the calculation:
`If (CNSAudit_Version = "?" ; 0 ; 1)`



Step 9.

Step 10.

Step 11.

10. Uncheck the Validate only if field has been modified option near the bottom of the Specify Calculation dialog.
11. Press OK to close the Specify Calculation dialog.
12. *Optional:* Back on the Options for Field dialog, check the Display custom message if validation fails option and enter in something similar to:
Modification to this database has been disabled because changes cannot be recorded to the Audit Trail log. Please make sure CNS Audit has been installed and enabled.
13. Press OK to close the Options for Field dialog.
14. Repeat steps 3-13 for each table that has been set up for auditing.
15. Close the Define / Manage Database dialog.