# EMR - Electronic Medical Records

This document applies to Raintree version 10.2.500
Revised 07/15/2014

**Raintree Systems, Inc.**
27307 Via Industria
Temecula, CA 92590
Tel: 951-252-9400
Fax: 951-252-9401
www.raintreeinc.com

# Contents

## Templates and Categories

## EMR Lists

Template Editor

# Advanced

## Electronic Medical Records

# Electronic Medical Records

EMR records provide a HIPAA-compliant means of tracking clinical documentation with full rich-text formatting support. In addition to built-in fields defined for each record, it is a simple matter to assign custom fields and screens for any EMR through the use of EMR templates. In addition, EMR records can trigger scripts that display built-in or customized screens, run reports, send email messages, schedule appointments, add charges to the ledger, or perform any other task that can be accomplished via scripting.

The EMR system consists of three components:

- **EMR templates** define the structure of EMRs.
- **EMR records** are records created by running and filling an EMR template. EMR records are always linked to a patient account.
- **Narratives** are the rich-text documents that accompany EMRs. Narratives allow merging information from the EMR into a printable document.

All EMRs have a basic set of fields that are filled in automatically - owner, date and time when the record was created, patient number, EMR template code and category. These are present and filled in all EMRs. Provider, location and case number are not required. Most EMRs also contain custom data entered through a screen - patient medications, progress notes, etc.

EMRs support document versions, also known as **amendments**. The entire amendment history for a document is available within Raintree. There is no practical limit for the number of times a document can be amended, nor the type of changes which can be made to an EMR document.

Another advantage of EMRs is the support for electronic signatures. Users can sign EMRs using their passwords. Once the password is successfully entered, the EMR record and its attached RTF document become read-only; a signed-off progress note, for example, can never be altered. Note that you can still amend the record while keeping the original intact.

When you add an EMR, data from a previous EMR of the same type may be loaded into the new EMR. This process is known as **rollforward**. Rollforward speeds up creating regular documents, for example progress notes or measurements - you will have the previous record's data on the screen and only need to modify the data that has changed.

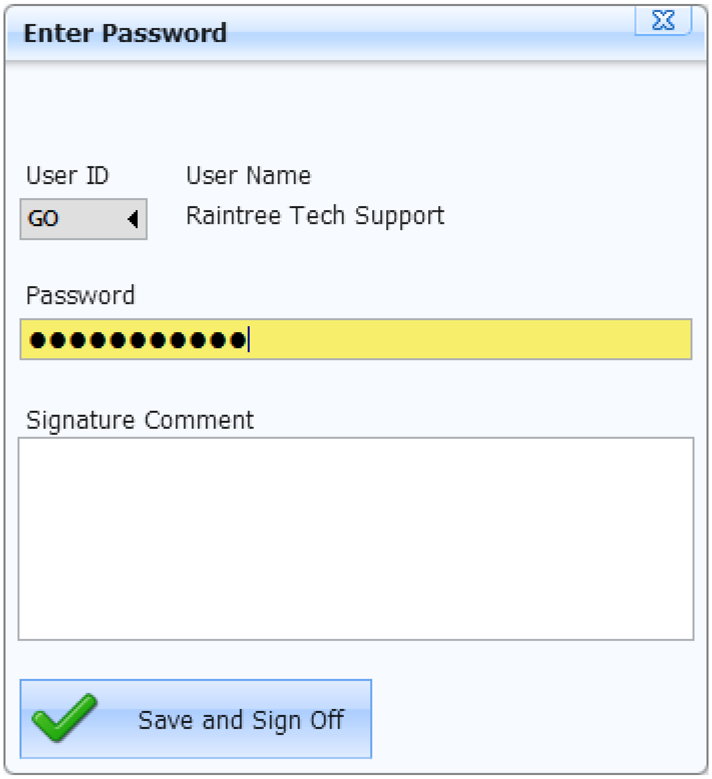
# EMR sign-off

When you sign off an EMR, you confirm with your password that the document is completed. If you want to make any changes to the document after signing it off, you need to amend the EMR.

There are three ways to sign off a single document:

- In the EMR editor, click **Save and Sign Off** in the bottom right corner of the screen.
- Right-click on a record in the **Dashboard** and select **Sign Off** to open the record in the EMR Editor.
- When viewing a list of EMR records, press **Enter** on the EMR you want to sign off, and select **Sign Off/Complete** in the menu that appears.

You are always prompted for your password, when signing off records. You are only required to fill out the **Signature Comment** field if you are signing off an amendment.



Enter Password

User ID: GO

User Name: Raintree Tech Support

Password: ●●●●●●●●●●●●

Signature Comment:

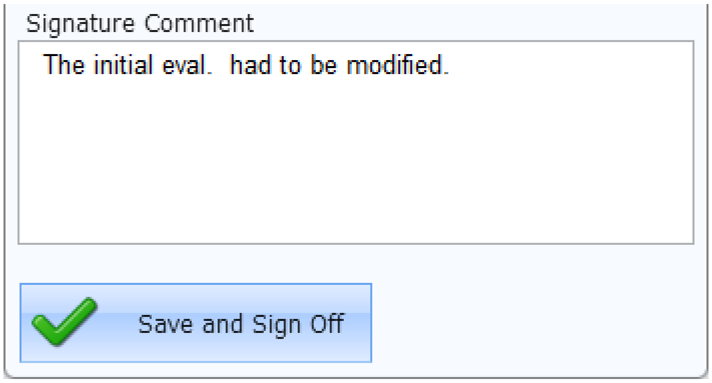
- Click **Save and Sign Off** to open the record's narrative and save, mark it as reviewed or sign it off.

Additional scripts may extend the sign-off functionality. These are specified in the EMR template setup.

**Note**: Only the record owner can sign the record off. If you are the record owner and you want to sign the record off, then you do not have to be logged in as the record owner. Raintree prompts for the record owner's password and you can sign the record off, while being logged in as another user.

## Signature comment

You can add a comment to your signature in the password entry screen, when signing off a document. This allows you to provide an explanation for any amendments you have made.



Signature Comment

The initial eval.  had to be modified.

Although the **Signature Comment** field is always visible when you are prompted for your password during sign-off, you are only required to enter a comment when making an actual amendment. If you simply sign off or cosign an EMR, then you can proceed without entering a comment.

When you amend a document, you are first prompted for an amendment reason. This amendment reason carries over to your signature comment. Thus you can review and modify your initial amendment reason if you find that you have made additional changes.

You can define the minimum signature length in the global.ini settings. The signature comment does not roll forward to other records.

# EMR Amendments

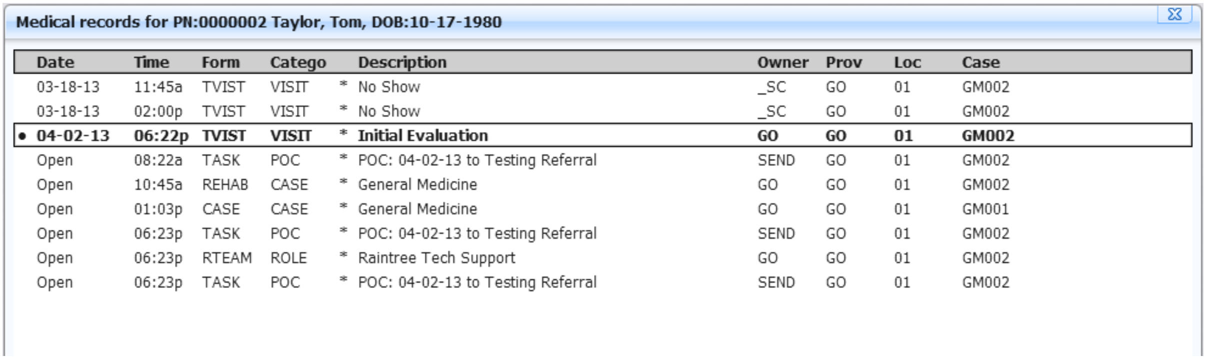
The Amendment process allows users to track different versions of signed-off documents. Once a document is signed off and made read-only, users can add amendments to that document. Each amendment is a new copy of the previous version of the EMR document, with its own control fields, RTF document and user-defined fields. Since amendments are themselves EMR documents, each is signed off and can be amended.

Adding an amendment does not change the EMR record being amended, it changes a copy of that record. The full **amendment history** of a document is stored within the EMR table, allowing access to all the different versions of a single entry, such as a Progress Note. There is no practical limit for the number of times a document can be amended, nor the type of changes that can be made to an EMR document.

You can amend a document in any list of EMR records.

1. Press **E** on the record. The **Completed EMR** menu is displayed.
2. Select **Amend / Correct**. The Amend / Correct menu is displayed.
3. Select the appropriate option - **Start Over** to modify the template settings (location, provider, etc.); **Modify** to re-open the input screen; or **Text Only** to modify only the merged document.

If an EMR has any amendments, the record will have a small dot in the first column.



Medical records for PN:0000002 Taylor, Tom, DOB:10-17-1980

| | Date | Time | Form | Catego | | Description | Owner | Prov | Loc | Case |
|---|---|---|---|---|---|---|---|---|---|---|
| | 03-18-13 | 11:45a | TVIST | VISIT | * | No Show | _SC | GO | 01 | GM002 |
| | 03-18-13 | 02:00p | TVIST | VISIT | * | No Show | _SC | GO | 01 | GM002 |
| • | **04-02-13** | **06:22p** | **TVIST** | **VISIT** | * | **Initial Evaluation** | **GO** | **GO** | **01** | **GM002** |
| | Open | 08:22a | TASK | POC | * | POC: 04-02-13 to Testing Referral | SEND | GO | 01 | GM002 |
| | Open | 10:45a | REHAB | CASE | * | General Medicine | GO | GO | 01 | GM002 |
| | Open | 01:03p | CASE | CASE | * | General Medicine | GO | GO | 01 | GM001 |
| | Open | 06:23p | TASK | POC | * | POC: 04-02-13 to Testing Referral | SEND | GO | 01 | GM002 |
| | Open | 06:23p | RTEAM | ROLE | * | Raintree Tech Support | GO | GO | 01 | GM002 |
| | Open | 06:23p | TASK | POC | * | POC: 04-02-13 to Testing Referral | SEND | GO | 01 | GM002 |

## Amend / Correct

This menu is displayed when you **amend an EMR**. The following options are available:

**Modify**
Reopen the input screen.

**Text only**
Modify only the merged document.

**Start over**
Modify the template settings (provider, location, etc.).

**VOID**
Mark the record voided.

You can also select **Strike over previous** to strike over the previous version of the edited text. This is useful to point out the changes you make.

The **Amendment Reason** field at the bottom of the screen allows you to provide an explanation for the amendment. You can review and edit this comment after having modified the document, during the sign-off process. You must enter a reason for the amendment in order to sign it off.

## Voiding EMR records

You can void EMR records, if you discover that some records are not valid and need to be disabled. If you void an EMR record, then the record is not deleted; this enables keeping track of the amendment history.

**Note**: You can only void tickets that have not been posted.

You can void only the last signed off record in the amendment chain; all the records in the chain are automatically voided, too. For example, if the record has been modified four times, then you can only void the fourth amendment; all four records are marked as void.

Note the following aspects about voided EMR records:

**Visual changes**

"VOID" is added to the beginning of the amendment chain's last record description; and the amendment type of the last record is set to "VOID" in the amendment history.

**Rollforward**

All the amendment chain's EMR records are disabled from being rolled forward. The previous record before the amendment chain records is rolled forward, instead. Voided records are not displayed in the rollforward list.

**Narrative**

A built-in, but customizable, narrative is attached to voided records, indicating when and by whom the record was voided. The original narrative is not merged to the new one.

**Non-amendable**

You cannot amend voided records.

**Non-reversable**

You cannot un-void records; you have to create a new record, if you want to restore a voided record.

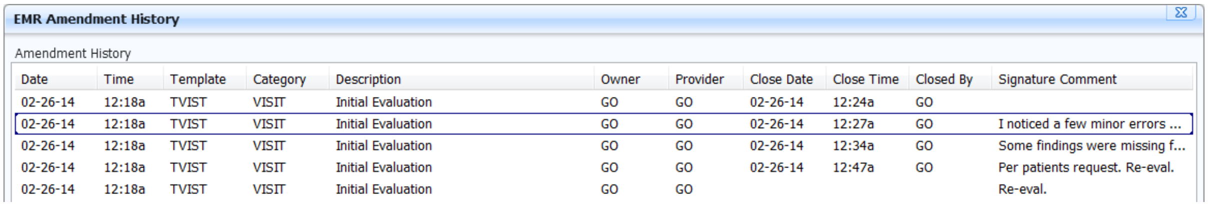
## EMR Amendment History screen

You can open the **EMR Amendment History** screen by pressing **Enter** on any EMR and selecting **View Amendment History**.

From this screen you can find information on amendment history and amendment requests, and request for an amendment to be made.

## Amendment History list

The **Amendment History** list displays the chain of amendments that have been made to the selected EMR. You can find information concerning the date, time, template, category, owner, signing off, and the reason for the amendment from this list. Both open and closed amendments are displayed. If you selected an EMR that has no amendments, then only the original record is displayed.



EMR Amendment History

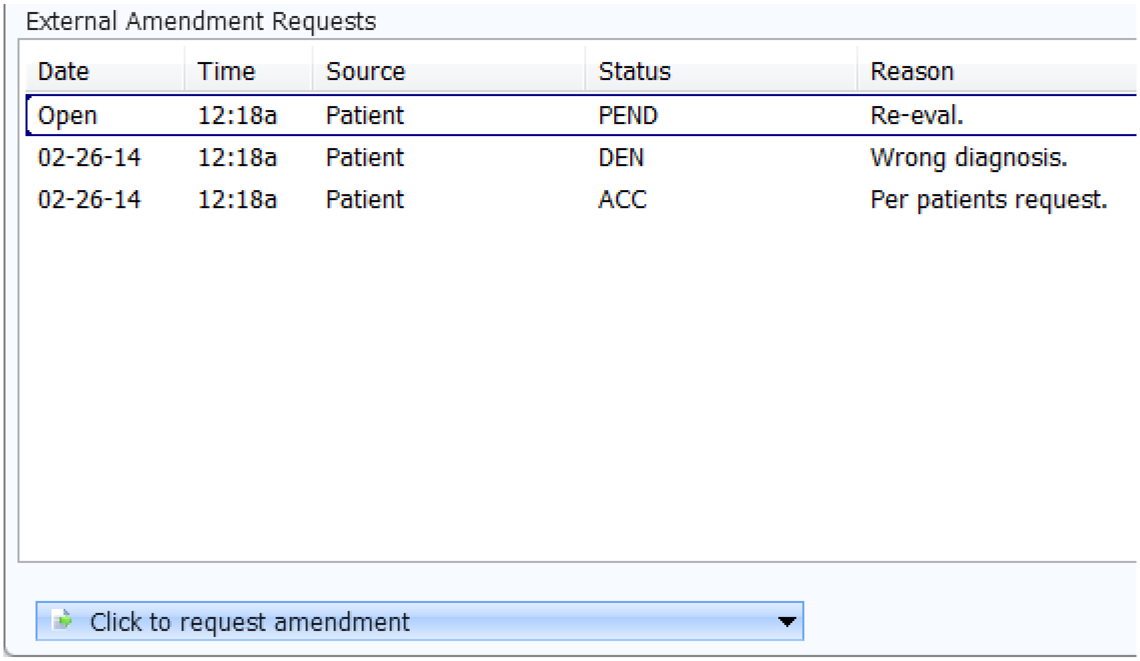
Amendment History

| Date | Time | Template | Category | Description | Owner | Provider | Close Date | Close Time | Closed By | Signature Comment |
|---|---|---|---|---|---|---|---|---|---|---|
| 02-26-14 | 12:18a | TVIST | VISIT | Initial Evaluation | GO | GO | 02-26-14 | 12:24a | GO | |
| 02-26-14 | 12:18a | TVIST | VISIT | Initial Evaluation | GO | GO | 02-26-14 | 12:27a | GO | I noticed a few minor errors ... |
| 02-26-14 | 12:18a | TVIST | VISIT | Initial Evaluation | GO | GO | 02-26-14 | 12:34a | GO | Some findings were missing f... |
| 02-26-14 | 12:18a | TVIST | VISIT | Initial Evaluation | GO | GO | 02-26-14 | 12:47a | GO | Per patients request. Re-eval. |
| 02-26-14 | 12:18a | TVIST | VISIT | Initial Evaluation | GO | GO | | | | Re-eval. |

Pressing **Enter** on an EMR opens a menu which allows you to view the document and create additional amendments.

## External Amendment Requests list

The **External Amendment Requests** list displays a list of all task records that have the selected EMR linked and have Request Record Amendment as their task type. These include all accepted, denied and pending amendment requests that have been made for the selected EMR.



External Amendment Requests

| Date | Time | Source | Status | Reason |
|---|---|---|---|---|
| Open | 12:18a | Patient | PEND | Re-eval. |
| 02-26-14 | 12:18a | Patient | DEN | Wrong diagnosis. |
| 02-26-14 | 12:18a | Patient | ACC | Per patients request. |

You can edit an amendment request that is still pending, but once a request has been accepted or denied it becomes read-only.

The following information is displayed in the list:

**Date**

The date when the amendment request was accepted or denied. If the request is still pending, the date is displayed as **Open**.

**Time**

The time when the amendment request was made.

**Source**

The person who requested the amendment.

**Status**

The request status, indicating whether the request has been accepted or denied, or whether it is still pending.

**Reason**

The explanation for the amendment request.

## Amendment requests

You can make an amendment request if you need to have a record amended or corrected and you are not the owner of the EMR.

Making an external amendment request creates a new task record that will be sent to the owner of the EMR. The task record will have the selected EMR linked to it, and its task type will be set to Request Record Amendment.

As you complete the amendment request, the owner of the record is able to review your request. Based on the information you provided, the request can either be accepted or denied. When an amendment request is accepted, a new amendment is automatically created for the linked EMR.

You can make and view amendment requests for a specific EMR from its **EMR Amendment History** screen.

## Requesting amendments

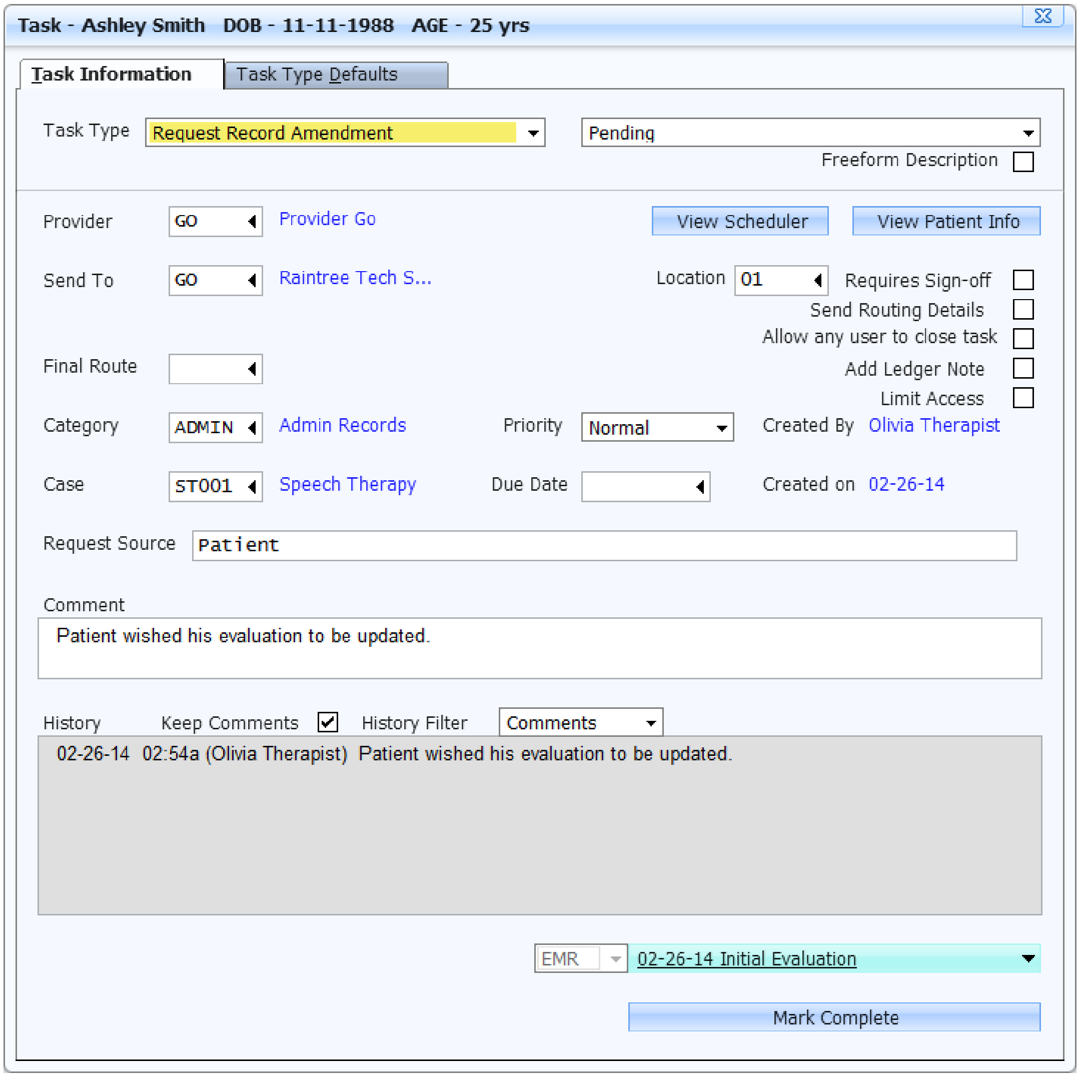
You can request an amendment to be made to an EMR from the **EMR Amendment History** screen.

**Caution**: When you exit the task record screen without completing the request, you will be prompted whether you also wish to delete the linked EMR.

To request an amendment:

1. Click the **Click to request amendment** button at the bottom of the screen.

   A new task record appears:



Task Type: Request Record Amendment | Pending
Freeform Description □
Provider: GO Provider Go

Send To: GO Raintree Tech S...
Location: 01
Requires Sign-off □
Send Routing Details □
Allow any user to close task □
Final Route:
Add Ledger Note □
Limit Access □
Category: ADMIN Admin Records
Priority: Normal
Created By: Olivia Therapist
Case: ST001 Speech Therapy
Due Date:
Created on: 02-26-14
Request Source: Patient

Comment: Patient wished his evaluation to be updated.

History | Keep Comments ☑ | History Filter: Comments

02-26-14 02:54a (Olivia Therapist) Patient wished his evaluation to be updated.

EMR: 02-26-14 Initial Evaluation

2. (Optional) Modify the pre-populated task information if necessary.
3. Enter who requested the amendment, in the **Request Source** field.
4. Enter a comment explaining the amendment request.
5. Press **F10** to complete the request.

   The task record is sent to the owner of the EMR.

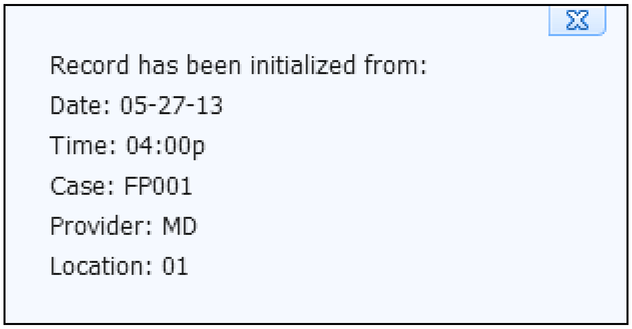
# EMR rollforward

## Overview

When you add an EMR and the patient already has an EMR with the same code, you may have the option to automatically load some data from the previous EMR to the record you're adding. This is referred to as EMR rollforward. The old record is called the source and the new record is the target of the rollforward.

Only **custom data** (the data you have entered) will be rolled forward; the EMR date, time, etc. are not copied. Also, some fields on the screen may not roll forward (for example the selected charges on a service ticket). This is achieved by adding those fields into a special widget group named `NoRollForward`. Fields synchronized via an EMR link widget are automatically excluded from rollforward.

When you roll an EMR forward, a notification may be displayed with the source record's data: date, time, case number, owner, and location. By default the notification is displayed as a hoverbox.



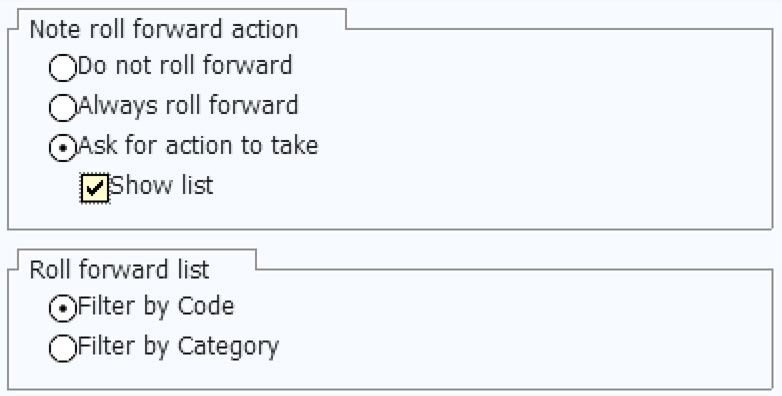
Record has been initialized from:
Date: 05-27-13
Time: 04:00p
Case: FP001
Provider: MD
Location: 01

You can select the notification type using the `global.ini` setting **RollForwardNotifyType**.

All rollforward attempts are logged and stored in the `logpath\rollforward` `[`*`lanid`*`].log` files. The log contains information such as the rollforward date and time, user, database, rollforward type, result, and the source and destination record data.

## Options

Rollforward is controlled by two settings on the **EMR Template** screen:



Note roll forward action
○ Do not roll forward
○ Always roll forward
● Ask for action to take
☑ Show list

Roll forward list
● Filter by Code
○ Filter by Category

The first setting determines the rollforward behavior. First option disables rollforward, second option forces the EMR to always roll forward, and the third option will prompt the user whether to roll the previous record forward or not. If the **Show list** checkbox is selected, a list of all suitable sources will be displayed; otherwise Raintree will attempt to select the best match for the target and roll it forward automatically.

The second setting applies to the list of possible rollforward sources. If set to **Code**, only records with the same template code will be included; otherwise Raintree will display all records with the same category.

## How is the record selected?

When looking for the best match for rollforward, Raintree first considers the code or category (depending on the template setting) of the current record - only records matching that will be available for rolling forward. Only the last matching record will be included in the list.

Next, Raintree compares the date and time of the source and target EMR - only records that have been added earlier than the target can be used as the source. In addition, the patient number of both records has to match.

The provider on the EMR must also match the owner (the user to whom the EMR is assigned). Editing the provider or owner on the EMR so that they will match will also trigger the rollforward process.

Finally, the case numbers of the target and the best possible match are compared. If the case numbers do not match, the list of all matches is displayed, allowing you to select the record.

**Note:** if the case numbers do not match, the list is displayed even if that option is not selected in the template setup. In this case a notification is also displayed.

## Re-rolling an EMR forward

If you have, by accident, chosen **not** to roll forward the previous EMR but you actually wanted to do that, then you have to delete the new record and start adding the EMR again.

When customizing a screen, you can add an Image Button named `_btnEmrRollForward` to the screen. Clicking this button will display the EMR re-rollforward list, allowing you to clear the values that were rolled forward, or to select a new source for rollforward.

# EMR security rights

There are several security rights related to EMRs.

The rights for administrating EMRs are:

- EMRPACK allows adding, editing, or deleting EMR plugins.
- EMRCAT allows adding, editing, or deleting EMR categories.
- EMRTEMPL allows adding, editing, or deleting EMR templates.
- EMRLIST allows adding, editing, or deleting EMR lists.

In addition, the DEBUG=E right is needed to view the plugin Settings section and system plugins.

You can also use **EMR access groups** to allow only specific users or user groups to access an EMR. You can also disallow specific actions (adding, editing, deleting, and so on) in EMR list setup.

When checking if the user is allowed to do something in an EMR list, Raintree checks for user permissions in the following order:

1. Is the action allowed in the EMR list setup?
2. Does the user have the rights for this LST file?
3. Does the user have access to EMRs in general?
4. Does the user have the rights for this particular EMR file?
5. Are there any EMR Access Groups defined for that specific EMR? If so, does the user belong to any of the specified groups?

## List-based security

When working with EMRs, you can specify the EMR list filename as a security function to control access to that EMR list:

- MYLIST.LST controls allowed actions in that particular EMR list (adding, editing, and deleting).
- MYPLUGIN:MYLIST.LST is similar to the previous example, but also adds the plugin name. You should always use this format to avoid problems when an EMR list with the same name exists in two plugins.
- MYPLUGIN:*.LST applies to all EMR lists in the specified plugin.

## Template / plugin-based security

When working with EMRs, you can also specify individual EMR templates and EMR templates inside plugins as security functions to control access to the templates. There are four ways of controlling access to EMR templates.

- MYEMR.EMR limits access to all EMRs with the specified EMR template. For example, EMRT2.EMR=AE allows only adding and editing all EMRs with the template EMRT2.
- MYPLUGIN:MYEMR.EMR limits access to specific EMR templates in the specified plugin. For example, EMRT1:EMRT2.EMR=AE allows only adding and editing all EMRs with the template EMRT2, if the template is located in the EMRT1 plugin.
- MYPLUGIN:*=AE specifies limitations to all files that are in the specified plugin. For example, EMRT1:*=AE allows only adding and editing files that are in the EMRT1 plugin.
- MYPLUGIN:*EMR/PCK specifies limitations to only EMR templates or picklists in the specified plugin. For example, EMRT1:*EMR=AE allows only adding and editing EMRs that are in the EMRT1 plugin. You cannot add/edit other files.

The above examples define security rights for adding or editing, but you can specify other security rights as well. These functions are not restricted to adding or editing only.

**Note**: Only users with E security right can open EMRs for editing. Users with other EMR security rights, but without the E security right, can view the EMR in read-only mode. For example, users with the EMRT2.EMR=F security right can only see the read-only EMR, while users with the EMRT2.EMR=FE security right can open the EMR and edit it.

The security rights work differently on the template and plugin level. This means that if the security rights for the individual EMR template are different from the template specified in the plugin, the security functions do not work at all. If there are similar rights defined for the individual template and the template in a plugin, then only the mutual part works.

For example, EMRT2.EMR=AE and EMRT1:EMRT2.EMR=DP. If the EMRT2 template has different security rights defined on the template level and the plugin level, the security functions do not work. However, EMRT2.EMR=AE and EMRT1:EMRT2.EMR=AD have one mutual security right. In this case only the mutual (adding) right functions for the EMRT2 template.

## EMR access groups

You can use **EMR access groups** to allow only specific users or user groups to access an EMR. Only the EMR owner and the users / user groups defined in **Access Groups** can see the EMR record. The record is hidden from the other users' lists.

By default, all EMRs are available for all users. You can define default access groups using the `DefaultAccessGroups` setting in the `global.ini`. The users or groups defined there are used as the default access groups for all EMRs.

You can also define how the users / user groups see or edit the specified EMR record in the list. This is controlled by the security function EMRACCESS=R/L.

- EMRACCESS=R enables you to open the EMR from the list as a read-only record. With this security right you can only read the specified EMR, but not edit it.

- EMRACCESS=L enables you to only see the EMR in your list, but not open it for reading or editing.

  If you try to open the specified EMR from your list, you receive a warning message stating that you do not have the rights for reading the specified EMR record.

If the specified EMR access groups have the security rights defined either way, then only the EMR owner is able to edit the EMR record. Users and user groups, that are not restricted with the security functions EMRACCESS=R/L, are also able to edit the specified EMR record.

You can define the EMR access groups from the **Medical Records** screen or from a specific EMR screen linked to the template. However, most EMR screens in Raintree do not have the **Access groups** field added by default. You can still add this field to an EMR from the Template Editor if necessary.

When adding the **Access groups** field from the Template Editor, you need to define the field name as `emrgroups` and the have the mask `&emrgroups:d1:30`.

## Defining EMR access groups

You can define EMR access groups from the **Medical Records** screen or from a specific EMR screen linked to the template. Either way, the functionality and workflow of defining EMR access groups is the same on the **Medical Records** screen as well as the EMR screen. The following example workflow explains defining access groups from the **Medical Records** screen.

From the patient's Medical Records list:

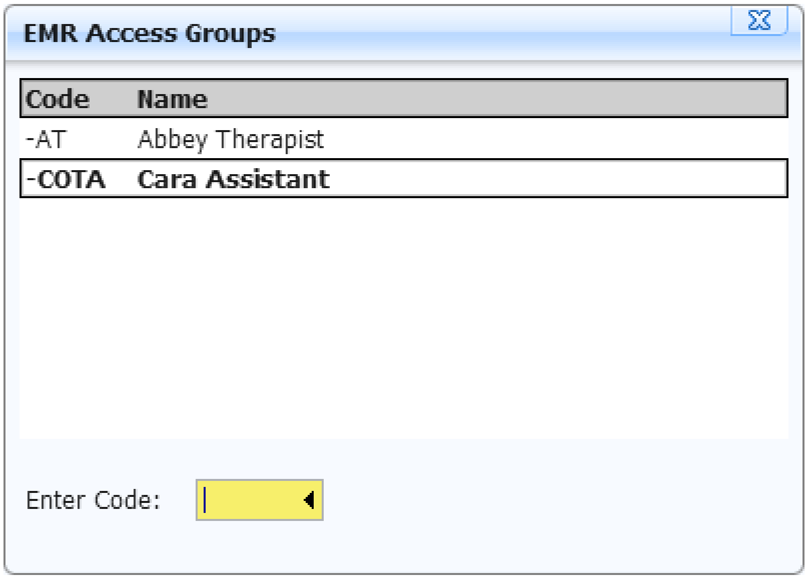
1. Press **A** to add.

   The **Medical Records** screen appears.

2. Define the necessary EMR options as usual.

3. Press **Tab** in the **Access groups** field.

   The **EMR Access Groups** screen appears.



| Code | Name |
|---|---|
| -AT | Abbey Therapist |
| -COTA | Cara Assistant |

4. Press **Tab** in the **Enter Code** field. Select the necessary user / user group and press **Enter**.

5. Repeat the previous step if you want to add more users / user groups.

6. Press **F10** to save the **Access groups** list.

   The access group users appear in the **Access groups** field in a comma-separated list.

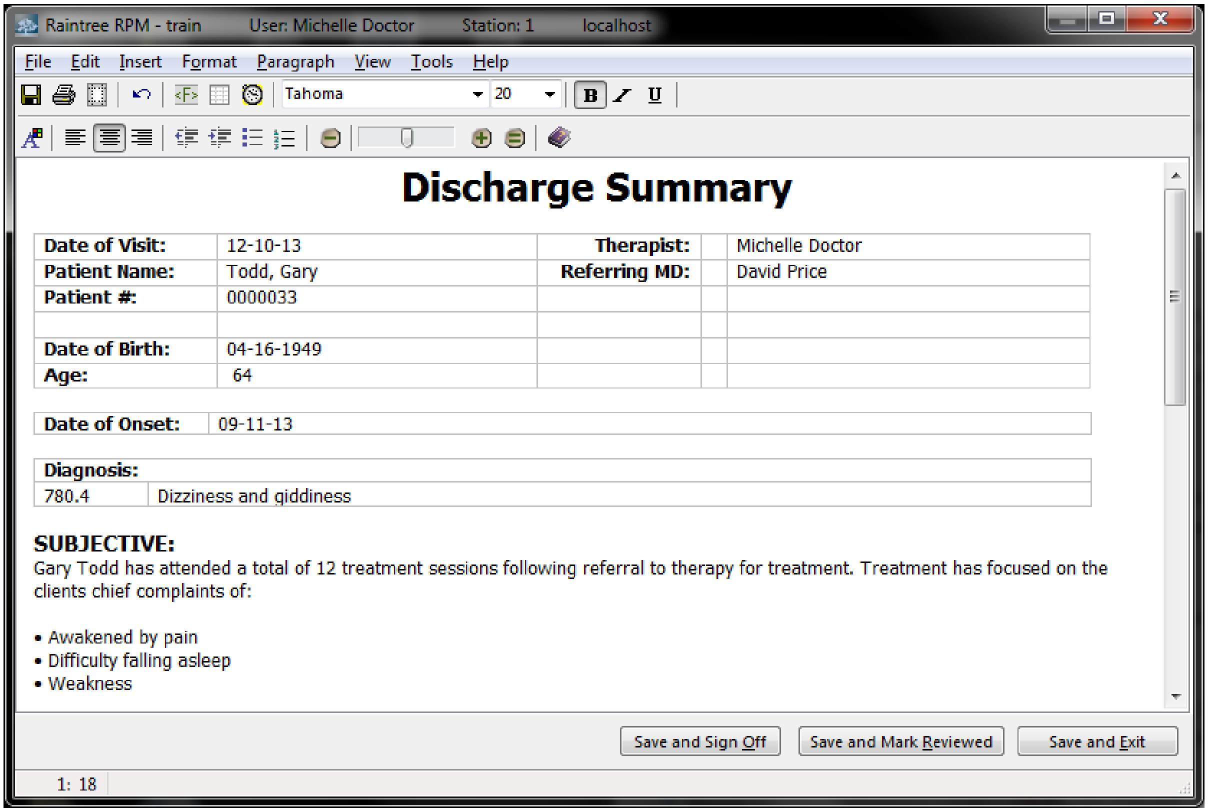
7. Press **F10** to save the base record and start defining the EMR itself.

Only the EMR owner and the users / user groups specified in the **Access groups** field are able to see the specified EMR record in their list. Depending on the users' security rights, they are either able to edit the EMR; open the EMR as a read-only record or only see the EMR entry in their list.

**Note**: Most EMR screens do not have a specific field for defining access groups. You need to add this field to the EMR screen from the Template Editor.

# EMR Editor

The EMR Editor is an enhanced version of the **RTF Editor**. It allows you to access and modify EMR narratives.



## Discharge Summary

| | | | |
|---|---|---|---|
| **Date of Visit:** | 12-10-13 | **Therapist:** | Michelle Doctor |
| **Patient Name:** | Todd, Gary | **Referring MD:** | David Price |
| **Patient #:** | 0000033 | | |
| | | | |
| **Date of Birth:** | 04-16-1949 | | |
| **Age:** | 64 | | |

| | |
|---|---|
| **Date of Onset:** | 09-11-13 |

| **Diagnosis:** | |
|---|---|
| 780.4 | Dizziness and giddiness |

**SUBJECTIVE:**

Gary Todd has attended a total of 12 treatment sessions following referral to therapy for treatment. Treatment has focused on the clients chief complaints of:

- Awakened by pain
- Difficulty falling asleep
- Weakness

There are several ways to open the EMR Editor:

- Create and save an EMR. If that EMR template has a narrative, it is automatically displayed in the EMR Editor.
- Preview the document while editing an EMR. You can do this from a button on the screen (if available) or from a list of EMRs.
- View the document of a signed-off EMR.
- Start amending an EMR in text-only mode.

The main area of the editor contains the merged narrative for the current EMR. You can insert additional merge tags; their content is merged when you save the document. You cannot edit an EMR that has been signed off.

Below the document are buttons to amend or save the EMR. The available buttons depend on whether the EMR has been signed off, as well as `global.ini` settings.

When editing an EMR:

**Save and Sign Off**

Closes the editor and starts the sign-off process. When you click this button, you are prompted for your password; if the password is entered correctly, the document is signed off and becomes read-only. A signed-off record cannot be edited - if any changes are required, they must be made in amendments of the document.

**Save and Mark Reviewed**

Closes the EMR Editor and marks the record as reviewed. You can sign off reviewed records as a batch without opening them.

**Save and Exit**

Saves the changes to the EMR document when work is still in progress. If this button is used to save and close the editor, you can still edit the EMR later.

When viewing a signed-off EMR:

**Amend**

Displays a menu where you can select how to amend the EMR. The options are the same as in the **Amend / Correct** menu.

Depending on the `ShowEditorAmendButton` `global.ini` setting, this button may be hidden.

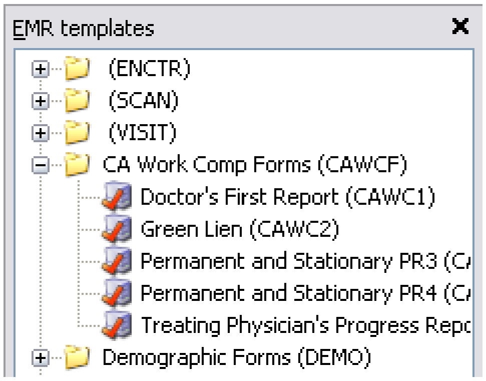
**Close Editor**

Closes the EMR Editor.

If the EMR has not been signed off, additional items may be displayed - list of all templates, tabs for individual sections in the document, and shortcuts to run individual templates.

## Templates pane

The **Templates** pane on the left contains a list of nested folders containing EMR templates. The folder tree is defined by EMR categories. Each folder can contain other folders and EMR templates. To view the items stored in each folder, click the plus sign (+) to the left or double-click on the folder name or icon to expand the folder. Double-clicking on any template within these folders will "run" the template. Depending on how the selected template was designed, this could insert text into the editor, run a script, prompt the user to select items from a list, display a popup window, and so on. Templates appear in the list with a red checkmark to the left, followed by the template description and the template code in parentheses.

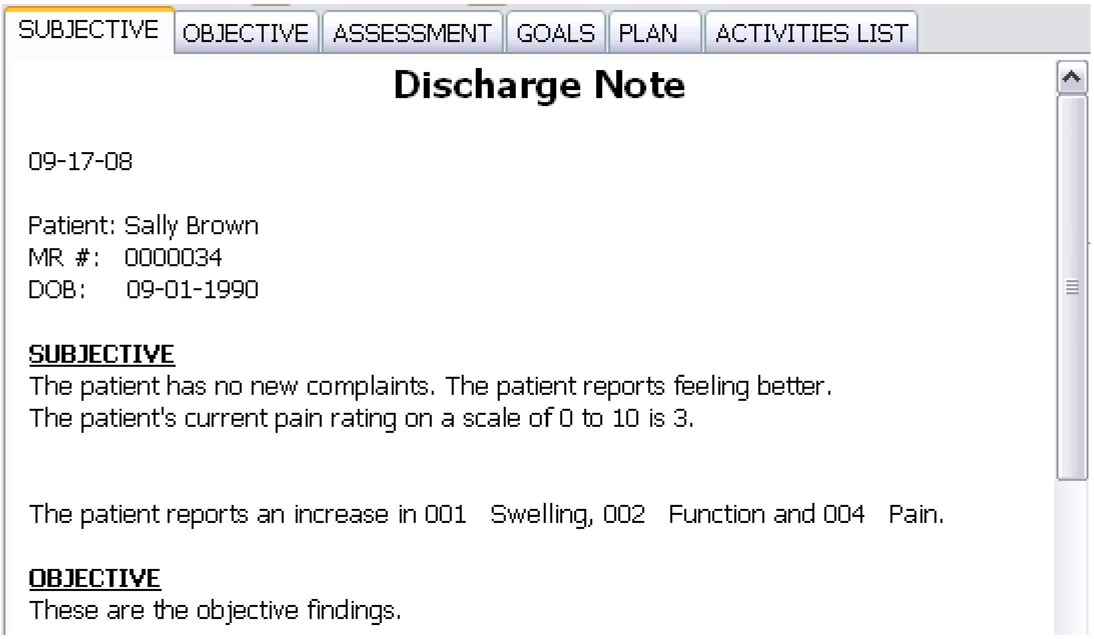

EMR templates
×
(ENCTR)
(SCAN)
(VISIT)
CA Work Comp Forms (CAWCF)
Doctor's First Report (CAWC1)
Green Lien (CAWC2)
Permanent and Stationary PR3 (C
Permanent and Stationary PR4 (C
Treating Physician's Progress Repo
Demographic Forms (DEMO)

To run a template, you can also type the code for that template into the **Run** field at the bottom of the templates frame and press **Enter**. For a deeply-nested template, this can be much faster than expanding out each folder to find the desired template.

To show the **Templates** pane, press **Alt+L** or open the **Tools** menu and select **EMR Templates**. To hide it, click the "X" in the top right corner of the Templates pane.

## Sections

EMR documents can contain "sections," defined by the templates used for creating the EMR documents. In the following example the Discharge Note being edited contains six sections, only the first two of which are visible in the screenshot: "Subjective", "Objective", "Assessment", "Goals", "Plan", and "Activities List". These sections were defined in the EMR template used to create the sample Discharge Note document.



**Discharge Note**

09-17-08

Patient: Sally Brown
MR #: 0000034
DOB: 09-01-1990

**SUBJECTIVE**
The patient has no new complaints. The patient reports feeling better.
The patient's current pain rating on a scale of 0 to 10 is 3.

The patient reports an increase in 001 Swelling, 002 Function and 004 Pain.

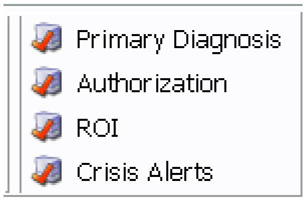
**OBJECTIVE**
These are the objective findings.

Each section corresponds to a tab above the text. When you click a tab, the cursor is moved to the beginning of that section and the document is scrolled to display that section. This can be helpful for jumping to a specific point within a very long document with many sections and sub-sections, such as an assessment.

One important use of sections within an EMR document is that they can be related to other EMR templates. You can set the sections up so that any text attached to templates located under a specific folder would be inserted into a specific section of the EMR document.

## Shortcuts pane

This pane is visible only if shortcuts have been defined in the EMR document's narrative using the `<shortcut>` merge command. Each shortcut will be displayed on the right as a button with a red checkmark. Clicking a shortcut button runs a template attached to that button.



# EMR plugins

EMR plugins is a way to organize and group EMR templates and related files. All files needed for an EMR template are stored in a single location.

Service ticket is an example of EMR plugins. A service ticket needs a screen where you can select items and enter data; an EMR template to allow signing off and amending the ticket; a narrative for printing; and scripts to provide additional functionality. Storing all these in a single location makes it possible to consider them as a single unit.

**Caution:** While most plugins have a specific purpose, there is also a generic plugin named "Default". Do not delete this plugin - it contains all non-plugin EMRs from the previous versions of Raintree, as well as templates and categories that were not added through a single plugin.

# View EMR Plugins

You can access the list of EMR plugins through the **More Tables** menu.

From the main menu:

1. Press **T** for **Tables**.
2. Press **M** for **More**.
3. Press **E** for **EMR Plugins**.

The list of EMR plugins is displayed. You can now select a plugin for editing or add a new plugin.

# EMR plugin list

On this screen you can manage EMR plugins and their components.

The screen contains the following tabs:

**Plugins**
All EMR plugins. You can select a plugin to view and edit its setup and component files (templates, scripts, etc.).

**Templates**
All EMR templates in all plugins.

**Narratives**
All narratives in all plugins.

**Screens**

All custom screens in all plugins.

**Scripts**

All scripts including setup options in all plugins.

**Setup Options**

All setup option scripts, for example case setup options.

**Note:** This tab is visible only when you select the **Show System Files** option at the bottom of the screen.

**Categories**

All EMR categories in all plugins.

**EMR Lists**

All EMR list definitions in all plugins. This list also includes the list definitions stored in the script path.

**Picklists**

All picklist definitions in all plugins.

**Overlays**

All overlays in all plugins.

**Menus**

All menus in all plugins.

Most tabs use color coding. System plugins and files have a cyan background. Other colors vary by tab:

- Inactive EMR plugins have a dark green background.
- Conflicting plugins (two active plugins with the same code) have a red background.
- PHP-controlled templates have a light green background.
- If a template with the same code exists both in the default plugin and a specific plugin, the template in the default plugin has green text on a white background.
- All scripts except setup options or system scripts have a purple background.

Below the tabs are located a button to check for changes in the EMR plugin folders, a button to configure EMR rules, and an option to toggle displaying system files.

## Searching EMR plugins

Searching the EMR plugins list is optimized to handle the possible duplicate plugin and file names.

When you press **F** in the EMR plugin list, the **Code** column is selected for searching by default. You can change the column to search by pressing **Up** or **Down**.

If there are multiple matches, Raintree focuses on the match in the active plugin. Other results are displayed below it, sorted by plugin code.

For example, if you have three EMR templates with the code ROLE in plugins ADM, MED, and ROLE, where ROLE is the active plugin, the results are displayed in the following order:

1. ROLE template in the ROLE plugin.
2. ROLE template in the ADM plugin.
3. ROLE template in the MED plugin.

## EMR plugin structure

EMR plugins are folders on the server for which a corresponding entry exists in Raintree. Each folder contains all needed files for a single plugin - scripts, screens, narratives, and so on. Most of the time, a script or screen is used only by a single EMR template, so they are stored together to simplify management. You can open a plugin and immediately view and edit the associated items.

All plugins are stored in `scriptpath/emr`. That folder contains a single subfolder for each plugin. That subfolder contains all plugin components, a file named `settings.ini` where the plugin settings and properties are stored, and a folder named `custom` if any component of the plugin has been customized.

To speed up working with plugins, the plugin settings and each component's properties are also stored in the `plugins` and `pluginfiles` tables in the database. Raintree automatically checks for changes in the plugin folders, but you can also check for plugin changes manually.

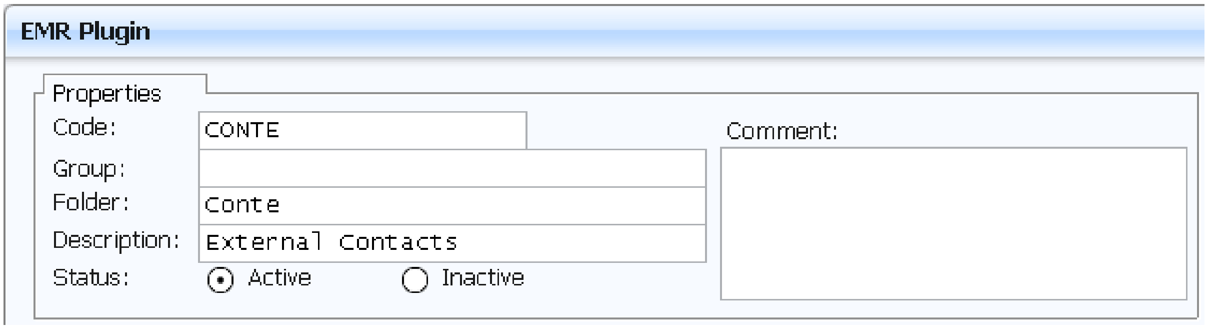
There's a difference between plugin codes and folders - while plugin folders are unique, you han have several plugins with the same code, each stored in a different folder. Only one of these plugins can be active at any time. This way you can create different versions of a plugin by using one while developing another. When the new version is finished, you mark the old one as inactive and the new one as active, thereby seamlessly switching over to the new version.

**Caution:** plugins should be managed only through Raintree. Specifically, renaming a plugin folder on the hard drive manually may cause the plugin to stop working.

When you edit files in EMR plugins, a backup of the edited files is created in `logpath/FileHistory/`*`database`*`/emr/`*`[Plugin folder]`* . This backup is created only if you edit the files via Raintree. If you edit the files manually (for example, using Notepad), the backups are not created.

# EMR plugin properties

An EMR plugin is defined through its properties.



EMR Plugin

Properties

Code: CONTE

Group:

Folder: Conte

Description: External Contacts

Status: ● Active ○ Inactive

Comment:

**Code**

Plugin code. Does not have to be unique; however, only one plugin with the given code can be active at any time.

**Group**

A group for the plugin.

Groups are used only for sorting plugins on the main EMR plugin list. Entering the same group name for related plugins will make them easier to find.

**Folder**

Name of the folder in `scriptpath/emr` where the plugin files are stored.

**Description**

Description of the plugin.

**Status**

Indicator whether the plugin is active.

**Inactive plugins** are effectively hidden from the users - templates in those plugins cannot be used to create EMRs, etc. Any EMRs created while the plugin was active will continue to work.

**Tip:** plugins that are still being developed should be marked inactive.

**Comment**

A free-form optional comment about the plugin.

# EMR plugin settings

EMR plugins have four settings that allow you to configure access to the plugin. These settings can be configured in the **Settings** section on the **EMR Plugin** screen.

**Allow copying**
If set, you can create a copy of this plugin.

**Allow versioning**
If set, you can create another version of this plugin.

**Allow customizing**
If set, you can customize this plugin.

**System**
If set, the plugin is marked as a system plugin.

System plugins should not be edited by users and therefore are visible only if you have the DEBUG=E security right. Usually, script libraries are marked as system plugins - they contain a collection of functions and methods to support other plugins, but are not usable by themselves.

# Defining EMR plugin setup options

You can access EMR plugin setup options from the **EMR Plugins** list.

From the main menu:

1. Press **T** for Tables.
2. Press **M** for More.
3. Press **E** for EMR Plugins. The Electronic Medical Records table appears.
4. Select **Show System files** from the bottom right of the screen to display all system files.
5. Select the **Setup Options** tab.
6. Select the necessary plugin and press **Enter**. The setup options are displayed.
7. Press **Enter** on a setup option and define its values.

   The definitions may be linked to tables or already given as options in the setup.
8. Repeat step number 7 to define all necessary setup options.
9. Press **F10** to save the setup.

# EMR Rules

EMR rules are scripts that are executed when you access an EMR. There are two types of rules:

- **Basic EMR rules** apply to basic operations with an EMR - creating a new EMR, deleting an EMR, etc. These rules can be configured from the main **EMR Plugin** screen.
- **Plugin rules** apply to **other** areas of Raintree. They are commonly used to update the records in the plugin when data in a standard record changes. These rules can be configured from inside a plugin.

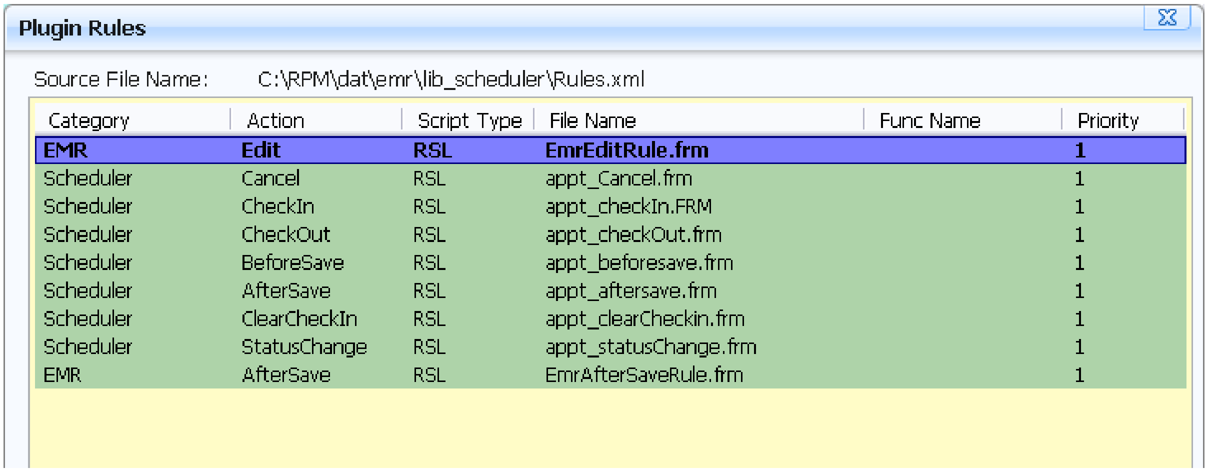
# Plugin Rules

In addition to the standard EMR rules, you can also define rules inside EMR plugins. Instead of applying to an EMR record, these rules are executed when you work in other areas of Raintree. Commonly, these rules are used to update the records in the plugin when a standard Raintree record is changed.

These rules allow EMR plugins to be self-contained - while the rules apply to other areas of Raintree, they are created and controlled by the EMR plugin. When you deactivate the plugin, the rules are deactivated as well.

These rules are stored in `Rules.xml` in the plugin folder. That file contains only the rules themselves, not the scripts that are executed.

To edit these rules, open the plugin and click the **Rules Setup** button in the bottom left corner of the plugin screen. A new screen opens, displaying a list of all rules defined in this plugin.



Plugin Rules

Source File Name: C:\RPM\dat\emr\lib_scheduler\Rules.xml

| Category | Action | Script Type | File Name | Func Name | Priority |
|---|---|---|---|---|---|
| EMR | Edit | RSL | EmrEditRule.frm | | 1 |
| Scheduler | Cancel | RSL | appt_Cancel.frm | | 1 |
| Scheduler | CheckIn | RSL | appt_checkIn.FRM | | 1 |
| Scheduler | CheckOut | RSL | appt_checkOut.frm | | 1 |
| Scheduler | BeforeSave | RSL | appt_beforesave.frm | | 1 |
| Scheduler | AfterSave | RSL | appt_aftersave.frm | | 1 |
| Scheduler | ClearCheckIn | RSL | appt_clearCheckin.frm | | 1 |
| Scheduler | StatusChange | RSL | appt_statusChange.frm | | 1 |
| EMR | AfterSave | RSL | EmrAfterSaveRule.frm | | 1 |

In the example above, most rules apply to appointment actions in the scheduler - canceling an appointment, checking an appointment in or out, etc.

You can press **E** to edit the setup of a rule, or press **S** to edit the script linked to the rule.

## Add a plugin rule

The EMR plugin rules are defined through EMR plugins. There are no limits to how many rules you can define; in addition, several plugins can apply a rule to the same category and action.

Before adding a rule you must set up the scripts that you want to execute when the rule is triggered.

1. Open an EMR plugin.
2. Click the **Rule Setup** button.

   The list of existing rules is displayed.
3. Press **A** to add a new rule.

   The **Rule Setup** screen is displayed.
4. Select a category for the rule.

   The category defines where the rule will apply.
5. Select an action for the rule.

   The action defines an action in that category where the rule will be applied.
6. Select the script type.

   Rules are usually RSL scripts.
7. Select the file type.

   You can choose between scripts in this plugin by using the dropdown next to the field.
8. (Optional) Select a function.

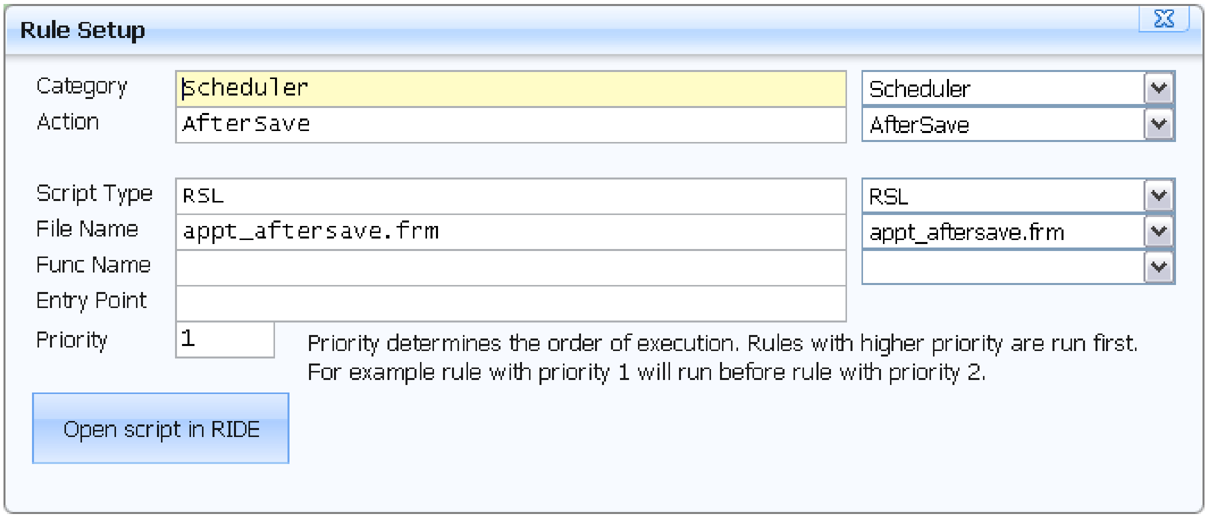
   If you want to run only a particular function of the script, enter its name here or select it from the dropdown next to the field.
9. Type the priority of the rule.

   Rules with a higher priority (smaller number) run before rules with a lower priority (larger number).
10. Press **F10** to save the rule.

The rule is set up. You can now add another rule or return to the EMR plugin screen.

## Plugin Rule Setup

On this screen you can edit an individual EMR plugin rule.



Rule Setup

| | | |
|---|---|---|
| Category | Scheduler | Scheduler |
| Action | AfterSave | AfterSave |
| Script Type | RSL | RSL |
| File Name | appt_aftersave.frm | appt_aftersave.frm |
| Func Name | | |
| Entry Point | | |
| Priority | 1 | |

Priority determines the order of execution. Rules with higher priority are run first. For example rule with priority 1 will run before rule with priority 2.

The options on this screen are described below. Next to most fields is a dropdown containing the available values.

**Category**

Area of Raintree where the rule will be applied. Corresponds to the name of a standard table.

The dropdown next to the field contains the most common tables.

**Action**

Action in the category where the rule will be applied. The dropdown contains various actions that can be used, for example opening a record, canceling an appointment, or logging in. These actions are not filtered by the category, so not all actions can be applied to all categories.

**Script Type**

Type of script that will run. Usually, the script type should be set to RSL.

**File Name**

Name of the script that runs when the rule is triggered. The dropdown next to the field contains all scripts in the current plugin.

**Func Name**

Function name from the selected script. Use this option if you want to execute only a particular function, not the whole script.

The dropdown next to the field contains all functions in the current script.

**Entry Point**

The entry point for PHP scripts.

**Priority**

Priority of the rule. Rules with a higher priority (for example, 1) will run before rules with a lower priority (for example, 2).

Below the options is also a button to open the selected script in Ride.

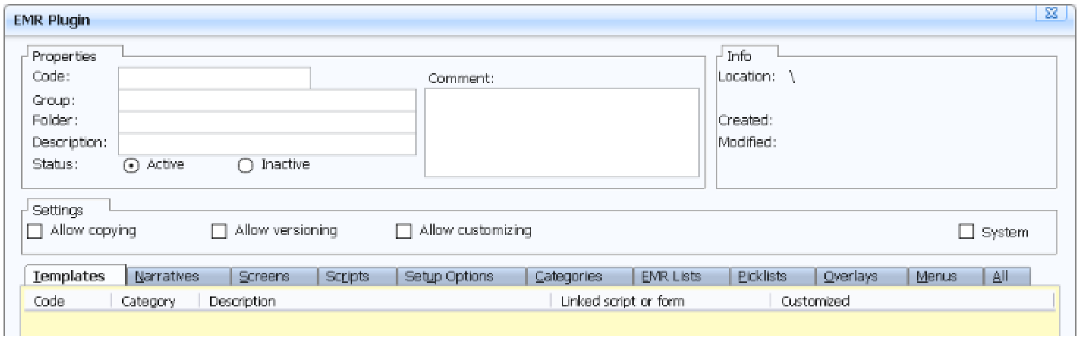
# Create an EMR Plugin

In most cases, new EMR plugins are created based on existing ones. However, you can also create an entirely new EMR plugin.

1. Open the list of EMR plugins.

2. Press **A** to add a new plugin.

   The empty **EMR Plugin** screen is displayed.



3. Enter the plugin code.

4. (Optional) Enter the group name.

   The group name is only used for sorting in the EMR plugin list.

5. Enter the plugin folder.

   The plugin files will be stored in this folder, located in `scriptpath/emr`.

6. Enter a description for the plugin.

7. (Optional) Set the plugin as inactive.

   When developing a plugin, you may want to set the plugin as inactive, so that incomplete templates will not show up when users are creating EMRs.

8. (Optional) Enter a comment for the plugin.

9. (Optional) Edit the plugin settings.

10. Press **F10** to save the plugin.

The plugin must be saved before you can add any items into it.

The plugin is created and appears in the EMR plugin list. You can now proceed with adding individual items to the plugin.

# Editing EMR Plugins

The standard Edit option allows you to rename the EMR plugin, edit its settings and properties, and edit the files inside a plugin. However, EMR plugins also have several advanced editing options.

In addition to editing the plugin, you can:

- **Copy the selected plugin**

  Use this option when you need several EMR plugins with similar functionality, for example, two different service tickets - one for inpatients and one for outpatients. The functionality of both tickets is the same, but the charges for in- and outpatients are different. Copying an EMR plugin will create a new plugin with a different code, so that both the original and new plugin are available.

- **Create a new version of the plugin**

  Use this option when you want to update a plugin, but keep existing templates and content working. For example, you want to create a new version of a service ticket for 2011, but keep the existing version as well. The new and old template have the same codes, but only one of the plugin folders will be active.

- **Customize the selected plugin**

  Use this option if you just want to have one plugin and do not need multiple versions, you should **customize the plugin**. All your changes will be made to the existing plugin. Customizing the plugin instead of editing the standard plugin files protects your customizations in case you download an updated version of the plugin - the standard files will be overwritten with the updated files, but your customizations remain untouched.

These options appear only if the respective permission is enabled in the EMR plugin settings.

## Copy an EMR Plugin

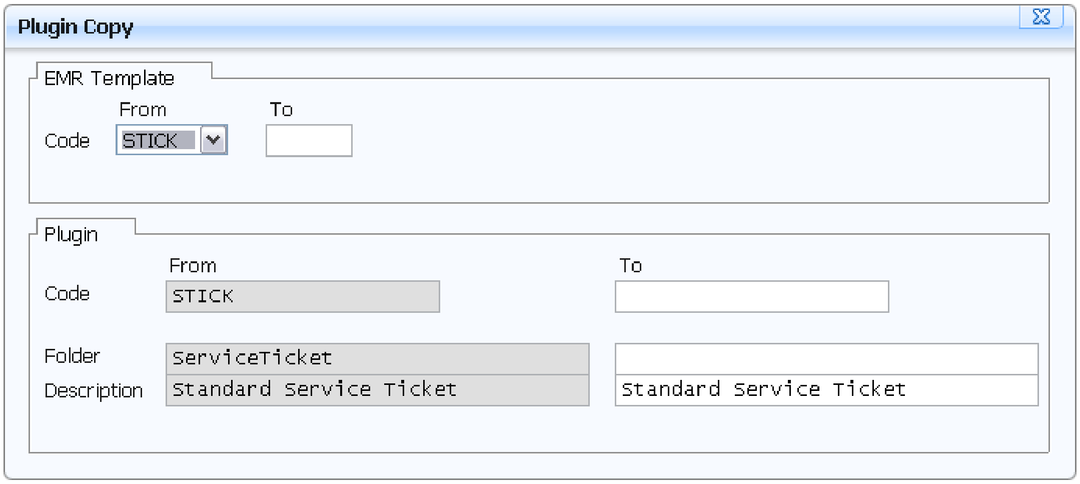
Copying an EMR plugin will create another copy of the plugin. Both plugins can be active at the same time.

**Note:** copying a plugin is possible only when copying is allowed in the plugin setup.

1. Open the list of EMR plugins.

2. Select the plugin that you want to copy and press **T**.

   A new screen opens.



3. (Optional) If the plugin contains an EMR template, enter the new code for the template in the **To** field.

   EMR templates have to be unique across the whole Raintree - two plugins may not contain a template with the same name. If you do not rename the template or the plugin contains more than one template, the new plugin will be marked as inactive. You can activate it only when you have renamed all templates in the new plugin.

4. Enter the new code for the plugin.

   When you entered a new code for a template, the new plugin code and folder fields are prefilled with the new template code.

5. Enter the new folder for the plugin.

6. (Optional) Edit the description for the plugin.

7. Press **F10** to create the copy.

A copy of the plugin is created. By default, it is also marked as active, unless there are some duplicate EMR templates. All screens, scripts, narratives, and EMR lists are also updated with the new code.

## Customize an EMR Plugin

Customizing an EMR plugin refers to customizing individual items inside a plugin. When you customize an item, a copy of it is created in the `/custom` subfolder of the plugin and this copy will be used from now on. Any future updates to the standard plugin will not overwrite the changes to the custom copy.

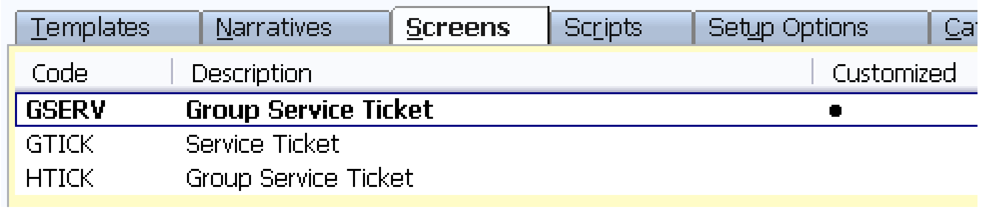
**Note:** customizing an item is possible only when customizing is allowed in the plugin setup.

To customize a plugin:

1. Open the plugin.
2. Select the tab for the item type you want to edit.

   For example, to edit an EMR list, select the **EMR Lists** tab.
3. Select the item you want to edit and either:
   - Press **U** to customize.
   - Press **Y** to customize based on a previously saved copy of a deleted customization.

A custom copy of the item is created. This is indicated by a mark in the Customized column.



| Code | Description | Customized |
|---|---|---|
| **GSERV** | **Group Service Ticket** | ● |
| GTICK | Service Ticket | |
| HTICK | Group Service Ticket | |

To revert to the standard version of the item, you can select the item and either:

- Press **L** to delete the customization.
- Press **O** to delete the customization but save a new copy of the customization in the `/custom` subfolder for later use.

## Creating a new version of an EMR Plugin

Creating a new version of an EMR plugin will create a new plugin with the same code. You can then edit the new version and activate the new version when you are ready.

**Note:** making a new version of a plugin is possible only when versioning is allowed in the plugin setup.

1. Open the list of EMR plugins.
2. Select the plugin that you want to use as the base for the new version and press **V**.

A new version of the plugin will be created and marked inactive. The folder name of the new plugin is the original's name with the suffix "copy". For example, when you copy a plugin in the folder `visitnote`, the new plugin will be stored in the folder `visitnotecopy`.

You can now develop the new inactive plugin while the older plugin is still used. When you want to switch to the newer version, you have to first mark the older version as inactive and then the newer version as active.

### Rename an EMR Plugin

Renaming an EMR plugin may refer to either changing the plugin code or folder. While you can change the code without major consequences, changing the plugin folder affects existing EMRs.

1. Open the list of EMR plugins.
2. Select the plugin that you want to rename and press **E**.
3. Move to the **Folder** field and enter the new folder name.
4. Press **F10** to save.
5. You are prompted to update any existing EMRs created using this plugin. Select **Yes** to continue.

The plugin folder has been renamed.

**Warning:** renaming the plugin folder manually on the server will render the existing EMRs from that plugin unusable!

## Delete an EMR Plugin

Deleting EMR plugins is not recommended because any EMRs created with that plugin will stop working. You should delete only plugins that have not been used; other plugins should be marked as inactive instead of deleting them.

1. Open the list of EMR plugins.
2. Select the plugin that you want to delete and press **D**.
3. Press **V** to verify deleting.

The plugin and all items it contained will be deleted.

## Adding Files to EMR Plugins

You should add files to an EMR plugin from the respective tab inside the plugin. Other ways of adding files are not recommended.

When you add a file from the EMR plugin list, it will be saved under the default plugin. This can be the desired result for common files, but in most cases the file is going to be used only in a very narrow context. In those cases, it is better to create the file directly inside a specific plugin.

When you add a file from the **All** tab inside a plugin, you have to select the file type first. The file will then be saved inside the current plugin. This results in an extra step (selecting the file type).

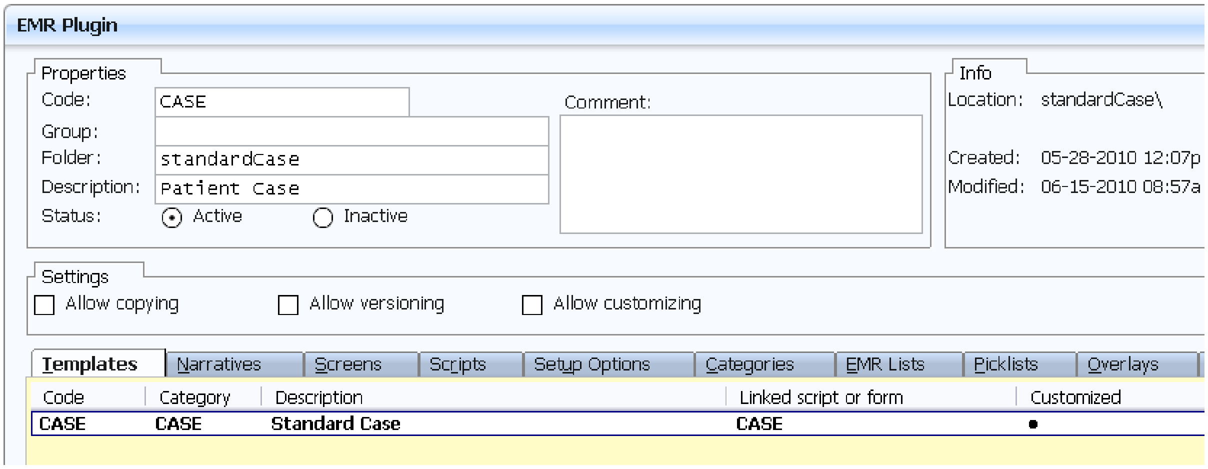
When you add a template or category through a lookup table (the table that opens when you press **Tab** on a screen to select an item), it will be saved under the default plugin. Again, in most cases this is not the desired result, so you should avoid creating templates on-the-fly.

## Templates and Categories

# EMR templates

EMR templates provide the structure for EMR records. You can link a template to a script, custom screen, or menu; also, you can specify a script for handling the sign-off of complex procedures. EMR templates often have a matching narrative that is used to print the EMR records created using that template.

EMR templates are stored in EMR plugins. The files are named `<code>.emr`; for example, the CASE template is stored as `CASE.emr`.



EMR Plugin

Properties

Code: CASE
Group:
Folder: standardCase
Description: Patient Case
Status: ● Active ○ Inactive

Comment:

Info

Location: standardCase\
Created: 05-28-2010 12:07p
Modified: 06-15-2010 08:57a

Settings

□ Allow copying □ Allow versioning □ Allow customizing

| Code | Category | Description | Linked script or form | Customized |
|---|---|---|---|---|
| CASE | CASE | Standard Case | CASE | ● |

Template names must be unique across all plugins - two plugins **may not** contain an EMR template with the same name.

You can make specific templates, which belong to active plugins, inactive; select the **Inactive** option in the **EMR Template** screen. This prevents users from adding records with this template, but retains the possibility to edit the existing records.

## Add an EMR Template

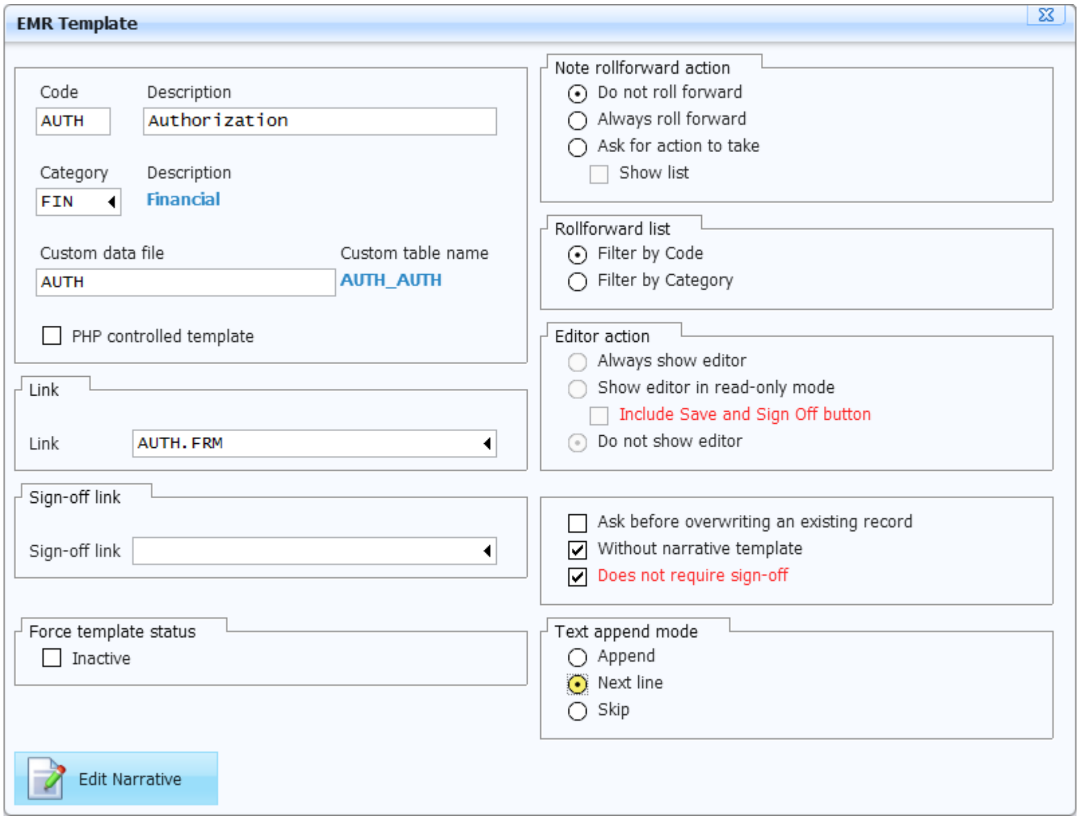
To create an EMR template:

1. Open an EMR plugin.

   The plugin opens with the focus on the **Templates** tab. If you already have the plugin open, you need to move to the **Templates** tab now.

2. Press **A** to Add.

   The **EMR Template** screen opens.



EMR Template

Code: AUTH
Description: Authorization

Category: FIN
Description: Financial

Custom data file: AUTH
Custom table name: AUTH_AUTH

□ PHP controlled template

Link
Link: AUTH.FRM

Sign-off link
Sign-off link:

Force template status
□ Inactive

Edit Narrative

Note rollforward action
● Do not roll forward
○ Always roll forward
○ Ask for action to take
□ Show list

Rollforward list
● Filter by Code
○ Filter by Category

Editor action
○ Always show editor
○ Show editor in read-only mode
□ Include Save and Sign Off button
● Do not show editor

□ Ask before overwriting an existing record
☑ Without narrative template
☑ Does not require sign-off

Text append mode
○ Append
● Next line
○ Skip

3. Type a code for the template.

4. Type the template description.

5. Select the EMR category.

6. (Optional) Edit the name of the custom data file.

   The custom data file defaults to the EMR category selected above. This is used for the custom table name, where the data for this template will be stored. The custom table name is *`datafile_template`* .

7. (Optional) Link the template to another item.

   Most templates require the user to fill a custom screen. To do that, you have to link a screen to the template. It is also possible to link other items; see **Linking items to an EMR template** for details.

8. (Optional) Define a sign-off script.

   If you want to define a special script to handle the sign-off procedure, type its filename with the path. Raintree also provides a signoff script that you can customize. See **Create a custom sign-off script for an EMR template** for details.

9. Set up editor and rollforward options.

10. (Optional) Create a narrative for the template.

11. Press **F10** to save.

## Copying EMR templates

You can make copies of EMR templates in the **EMR Template Copy** screen. The EMR template and related files (such as the NAR, XML, RTM files, scripts, setup options and overlays) are copied and saved with the new name defined in the **EMR Template Copy** screen.

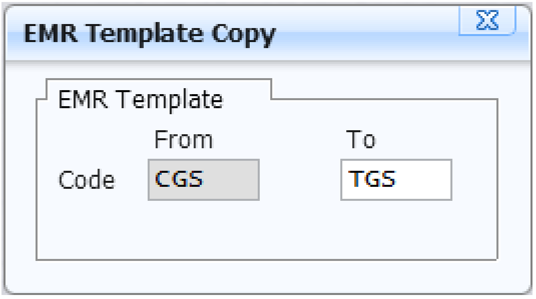
To copy EMR templates:

1. Open the list of EMR plugins.

2. Open an EMR plugin. The **EMR Plugin** screen opens.

3. Right-click on a template in the templates list.

4. Select **Copy** to copy the template. The **EMR Template Copy** screen appears.



EMR Template Copy

EMR Template

| | From | To |
|---|---|---|
| Code | CGS | TGS |

5. Type the new EMR template code into the **Code To** field.

6. Press **F10** to copy the EMR template.

A new EMR template is created with the defined name, consisting of the EMR template and the related files copied from another template.

## EMR Template Setup

You can control how the EMR template functions on the **EMR Template** screen.

**Code**

EMR template code. May contain only letters and numbers.

When adding a new template, then entering this code also automatically populates the file name in the **Link** field and the script name in the **Sign-off link** field. This speeds up your workflow and Raintree also verifies the consistency of linked file names and codes for you.

**Description**

Description for the template.

**Category**

Category for the template. Categories are used for sorting EMR templates.

**Custom data file**

The custom table where the template data will be stored.

**PHP controlled template**

Indicates that this template is controlled by PHP.

**Link**

An item that is run when the EMR record is created. Most templates have a custom screen linked to them, to allow entering data, but you can also link a menu or script.

When adding a new template, the file name in this field is populated automatically according to the defined template code. This way Raintree helps you keep template codes and linked script names consistent.

**Sign-off link**

Script to extend the standard sign-off functionality. When adding a new template, the link name is populated automatically for you.

**Text append mode**

How additional items are appended (this is used when combining multiple EMRs).

- **A - Append** - the text is inserted as normal.
- **N - Next line** - after the inserted text, a line break is added. This is the default option.
- **S - Skip line** - after the inserted text, an empty line is added.

**Note rollforward action**

How previous EMRs are rolled forward.

- **Do not roll forward** - when a new record of this type is added for a patient, all custom data fields (if any) will be blank. This is the default selection for a new EMR template.
- **Always roll forward** - when a new record using this template is added for a patient, the custom data values in the new record are pre-filled with the data from the most recent EMR record with the same template.
- **Ask for action to take** - when a new record using this template is added for a patient, you will be prompted whether the template should be pre-filled with the previous record's data.

When **Show list** is selected, a list of all available rollforward sources is displayed; otherwise, the latest suitable record is selected automatically.

**Rollforward list**

Which records are included for rollforward sources. By default, only records with the same template code are available for rollforward. This option allows you to match only the category of the source and target records.

**Editor action**

Determines whether to show the editor after the linked item has been completed (screen filled out and saved, script run, etc.):

- **Always show editor** - after saving the linked screen (or after the linked formscript has run), the EMR record is merged into an RTF document and the EMR editor is displayed. This is the default selection.

- **Show editor in read only mode** - the EMR template is merged, but the document is displayed as read-only. If the **Include Save and Sign Off button** is selected, you can sign off the EMR from the editor.

- **Do not show editor** - the editor will not be displayed. This can be used for EMRs that will be included in lists or other EMR templates, but never merged by themselves.

**Ask before overwriting an existing record**

If you edit an EMR that has not been signed off, the narrative will be re-merged into the RTF document. This will overwrite any changes that were made to the RTF. Select this option to prompt you before overwriting changes.

**Without narrative template**

Indicates that the template does not have a narrative.

**Does not require sign-off**

If selected, the template does not have to be signed off.

**Inactive**

Option to make the EMR template inactive. This prevents users from adding records with this template, but retains the possibility to edit the existing records.

**Edit Narrative**

Opens the associated narrative for editing. If the template does not have a narrative associated with it, this button does nothing.

**Create Narrative**

Opens the Template Editor for creating a narrative.

## Creating custom sign-off scripts for EMR templates

Raintree provides a library for developing sign-off functions for your EMR templates. You can customize the sign-off scripts for an EMR plugin or create new ones.

**Caution**: Any changes made to sign-off scripts will affect EMR templates that use the main sign-off library.

From the **EMR Template** screen:

1. Press **Tab** in the **Sign-off link** field.

   The field is populated with the link to the main sign-off library and the script opens in the editor.

2. Make any necessary changes to the sign-off script in the editor.

3. Save the changes to the sign-off script in the editor.

4. Press **F10** to save the changes to the template.

5. If you want to update existing EMRs with the new sign-off script, click **Yes** in the confirmation prompt.

   By default, the changes will only affect new EMRs. This prompt allows you to update existing EMR records with that sign-off script.

6. (Optional) Update other templates in the plugin.

   If you want to use the custom sign-off script in other templates as well, you need to open each one and change the sign-off link manually.

## Linking items to an EMR template

Most EMR templates have another item linked to the template. This item opens when you create the EMR record. You can specify the link in the **Link** section of the EMR template setup. The screen also displays a note if the specified file does not exist.

**Note**: When you are adding a new EMR template, the link is populated automatically according to the template code. You are still able to change the name.

You have five options:

- Type the file name with the path (for example, "cust.frm") to link an existing script.

- Type the menu name with the extension to link an existing menu.

- Type the screen's code to link an existing custom screen.

- Enter a template name to make the current template an extension of another template. The current template will then extend that one. All fields on this screen will populate with the values from the parent EMR, unless you have already edited them manually. The RTM and narrative of the parent will also be copied with the new code.

- Enter the name and (optionally) extension to create a new item, and press **Tab**. This creates a new file based on what you entered in the field and opens it for editing in the Template Editor. For example, if you entered "cust.frm", Raintree will create the file `cust.frm` in the current plugin and open it for editing.

**Note**: If you are adding a new EMR template, the name and extension in the **Link** field are already populated for you. In this case do not enter a new name but only press **Tab** to create the new file.

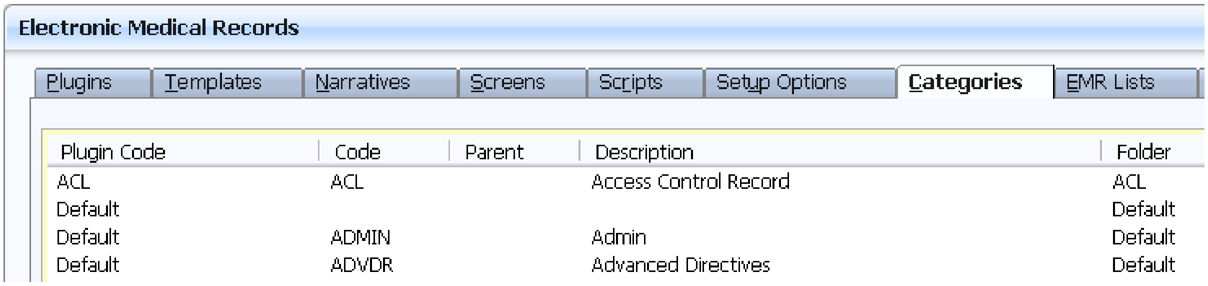
Remember that you can create a template that contains only a narrative. These templates do not need anything linked to them.

# EMR categories

All EMR templates must belong to a category. Categories help to organize the templates; for example, you may have several different visit records, but a single category for them. Categories are unlike EMR plugins in that a category is often used by several plugins. Category names must be unique across all plugins - two plugins **may not** contain an EMR category with the same name.

Categories are tree-structured, meaning that you can define child categories that are linked to a parent category. This is mainly useful when selecting templates in the EMR editor.

You can view the EMR categories from the EMR plugin list. The categories, including their parent category (if applicable) and location, are displayed on the **Categories** tab.



| Plugin Code | Code | Parent | Description | Folder |
|---|---|---|---|---|
| ACL | ACL | | Access Control Record | ACL |
| Default | | | | Default |
| Default | ADMIN | | Admin | Default |
| Default | ADVDR | | Advanced Directives | Default |

You can also view the categories inside a single plugin by selecting the **Categories** tab on the plugin screen.

## Add an EMR Category

Categories should be added through an EMR plugin. Most categories are stored in the default plugin.

1. Open the plugin where you want to add the category.
2. Click the **Categories** tab.
3. Press **A** to add a new category.
4. Type a code for the category.
5. (Optional) Specify a parent category if you want to create a linked child category.

6. Type a description.

7. Press **F10** to save.

# EMR Lists

EMR Lists are a way to define filtered interactive lists of EMR records without requiring any custom programming. These lists can be used as stand-alone table views ("live lists") linked from a menu option or drop-down field in a custom screen, as embedded lists or graphs within a screen (such as a dashboard), or as an RTF table that can be inserted into any EMR document. EMR lists can be very simple or very complex.

EMR Lists can be stored in EMR plugins or in the scriptpath. The files are named `<code>.lst`; for example, the VITAL list is stored as `VITAL.lst`.

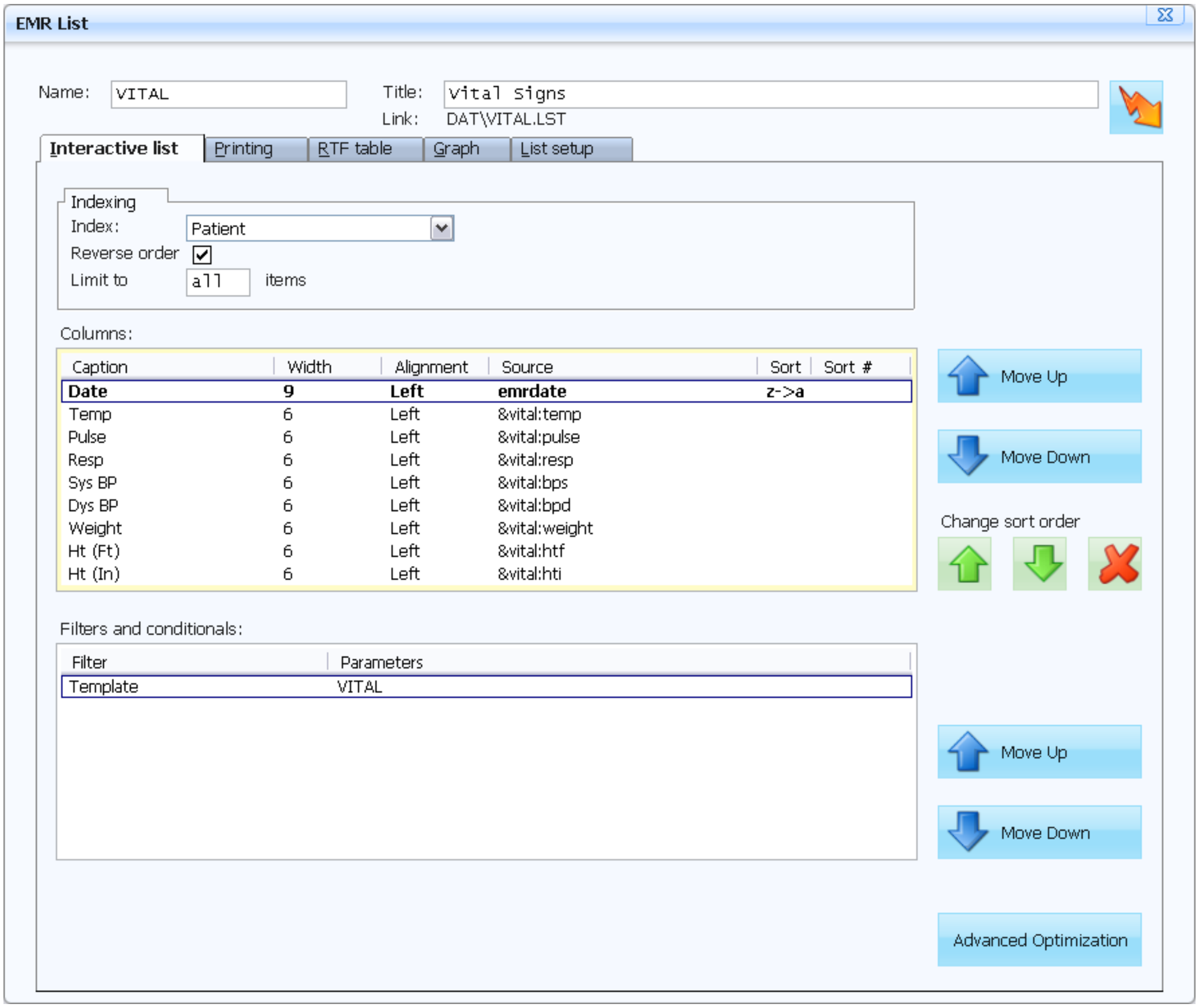
# Add EMR List

This is a basic overview of how to add an EMR list.

1. Open the plugin where you want to add the list.
2. Click the **EMR Lists** tab.
3. Press **A** to Add.
4. Enter a name for the list. The name may contain only letters, numbers and spaces.
5. Enter the list's description.
6. Select an **index** for the list.
7. Define the list **columns**.
8. Define the list **filters**.
9. Repeat for other tabs if needed.
10. Edit the **List setup**.
11. Press **F10** to save.

# EMR List screen

You can control how the EMR list functions on the **EMR List** screen.



EMR List

Name: VITAL

Title: vital signs

Link: DAT\VITAL.LST

Indexing

Index: Patient

Reverse order ☑

Limit to all items

Columns:

| Caption | Width | Alignment | Source | Sort | Sort # |
|---|---|---|---|---|---|
| Date | 9 | Left | emrdate | z->a | |
| Temp | 6 | Left | &vital:temp | | |
| Pulse | 6 | Left | &vital:pulse | | |
| Resp | 6 | Left | &vital:resp | | |
| Sys BP | 6 | Left | &vital:bps | | |
| Dys BP | 6 | Left | &vital:bpd | | |
| Weight | 6 | Left | &vital:weight | | |
| Ht (Ft) | 6 | Left | &vital:htf | | |
| Ht (In) | 6 | Left | &vital:hti | | |

Filters and conditionals:

| Filter | Parameters |
|---|---|
| Template | VITAL |

The **Name** field contains the name of the list. This name is used to identify the list. Similar to a typical Raintree code, it can contain numbers and letters, but unlike a code, you can also use spaces. Also, the name length is not limited.

The **Title** field contains the list description. It is only used in the EMR Lists table.

The **Preview** button is located in the top right corner of the screen. Clicking this will open a preview of the defined list, where you can verify that all column widths and filters behave as expected. Note that this preview will always show records for all patients, even if the list is indexed by patient.

Below the Name and Title fields are five tabs:

- **Interactive list** - the "base" list definition.
- **Printing** - customizes the list appearance when printed.
- **RTF Table** - customizes the list appearance when included in an EMR narrative using the `<rtftable>` tag.

- **Graph** - allows to display the list as a graph.
- **List setup** - controls how the interactive list behaves on adding and editing.

The first four tabs all have the same sections for the list **index**, **columns**, and **filters**. In the bottom right corner is the button to edit **Advanced Optimization** settings. In addition to that the Printing, RTF Table and Graph tabs have an additional button called **Copy from list**. Click this button to copy all settings (index, columns, filters and SQL) from the **Interactive list** tab to the currently selected tab.

**Note:** editing the **Printing**, **RTF Table** and **Graph** tabs is not required. If those tabs are blank, the settings on the **Interactive list** tab are used.

## Interactive List Tab

This tab contains the "base" list definition. The records included in this list can be embedded in a screen, linked to a menu option or listed in a drop-down field.

If nothing has been defined in the Printing and RTF Table tabs, this definition is used instead of them.

## Printing Tab

This tab defines how the EMR list will be printed when the user presses **P** in the list.

From this tab, you can also define a custom title and subtitle for the printout of the EMR list.

**Note:** you can use the **Copy from Interactive List** button to copy the setup from the interactive list. This way you can just edit the list and do not have to redefine all columns and filters again.

## RTF Table tab

This tab defines how a table of records is included in an EMR narrative. RTF tables can be inserted into EMR narratives (.NAR files) using the `<rtftable>` tag. For example if you have a list called VITAL, you can include it as a table by entering `<rtftable:VITAL.lst>` in the document. See **Using an EMR List in a Narrative** for details.

Note that you can use the **Copy from Interactive List** button to copy the setup from the interactive list. This way you can just edit the list and do not have to redefine all columns and filters again.

Below the list of filters is another field: **Display this text if list returns nothing**. This helps to set up a better narrative - if the list is empty, you can use this field to display a line of text instead. For example, if you have set up an EMR list to display the patient's previous visits, you can enter in this field "No previous visits."

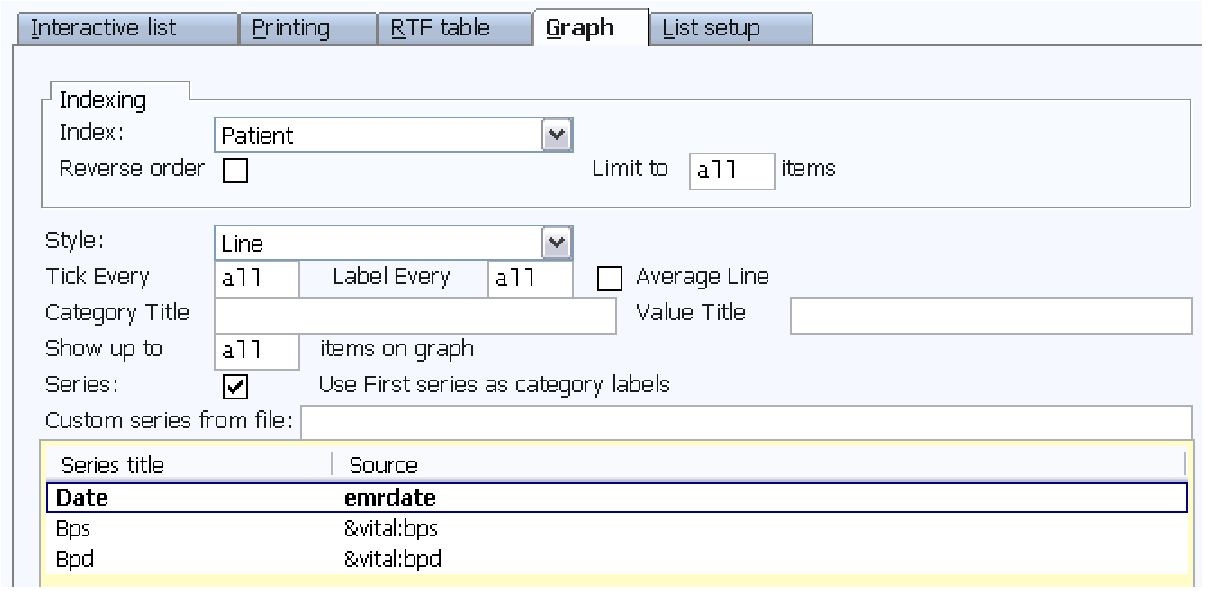
## Graph Tab

You can also set the list up to be displayed as a graph, for example to monitor changes in a patient's blood pressure during successive visits. Note that this tab has an additional section for defining how the graph will look.

You can use three styles for the graphs - Line (the default), Pie and Column. Each style has a different set of options. The graph can then be displayed on the screen or merged into a narrative.

**Tip:** if you need to display several graphs based on the same EMR, define new EMR lists for each graph and pull the data from the custom record of the original EMR.

For example, we would like to define a graph based on the vital signs that displays the patient's blood pressure changes. This is the graph setup:



Indexing
Index: Patient
Reverse order □
Limit to all items

Style: Line
Tick Every all
Label Every all
□ Average Line
Category Title
Value Title
Show up to all items on graph
Series: ☑ Use First series as category labels
Custom series from file:

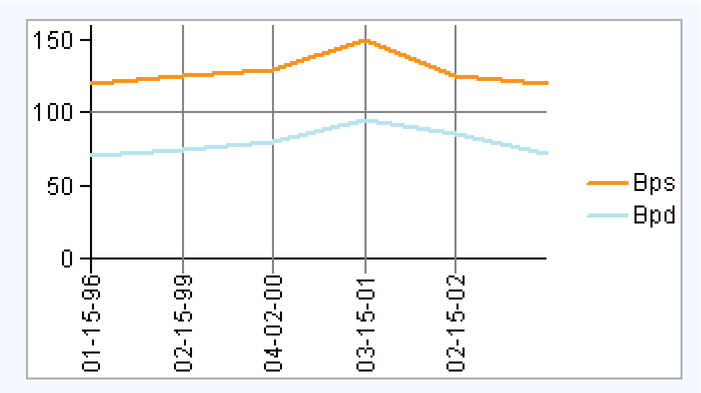
| Series title | Source |
|---|---|
| **Date** | **emrdate** |
| Bps | &vital:bps |
| Bpd | &vital:bpd |

The only filter used is for the VITAL template.

To show the graph on a screen, open the screen, create a **list box** and set the following properties:

- In **EMR List**, enter the name of the list that contains the graph, for example `BP.lst`.
- Select the **Show As Graph** option to display it as a graph; otherwise, it would be displayed as a regular list.

This is how the result will look:



When you right-click the graph, you can either select **Detail** to view the list data for the graph or **To Window** to display the graph in a separate window. In that window you can also copy the graph and paste it in Excel for example.

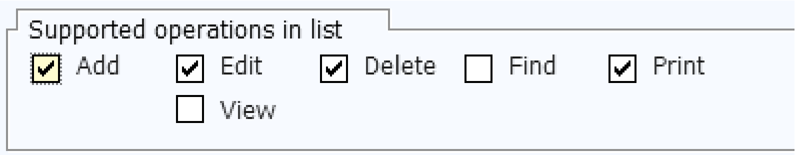
You can also use the graph in a narrative. The syntax is: `<graph:list.lst;width;height>`, for example `<graph:bp.lst;300;200>`.

## List Setup

On this tab, you can edit how the EMR list behaves on adding and editing.

### Supported operations in list

This section controls what the user is allowed to do in an EMR list.



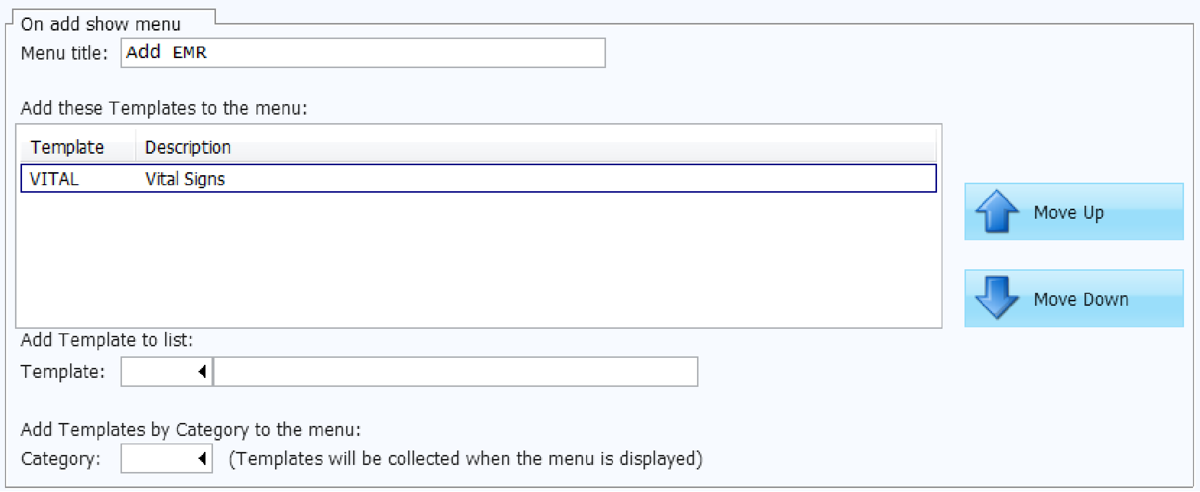
Supported operations in list

☑ Add ☑ Edit ☑ Delete □ Find ☑ Print
□ View

To allow an operation, select the respective checkbox.

### On add show menu

This section controls what happens when the user selects **Add** when viewing an EMR list.



On add show menu

Menu title: Add EMR

Add these Templates to the menu:

| Template | Description |
|---|---|
| VITAL | Vital Signs |

Add Template to list:

Template:

Add Templates by Category to the menu:

Category: (Templates will be collected when the menu is displayed)

If there is only one template defined in the **On add, run these templates** list, that template opens automatically. Otherwise, if there are multiple templates available, a menu is displayed where the user can select a template. You can also edit the menu title using the **Menu title** field and edit the available menu options.

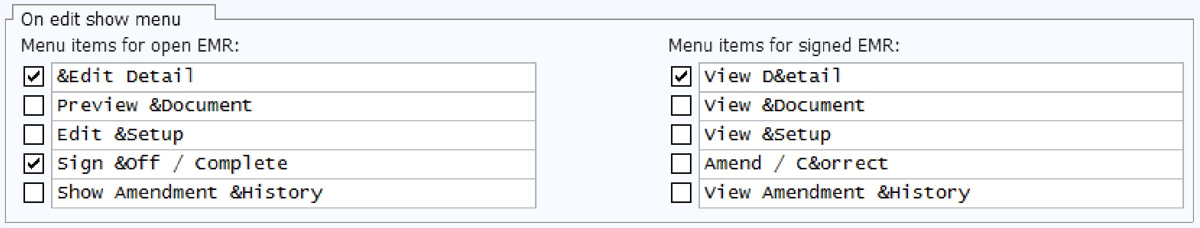
You can add templates to the list by entering a template code in the **Template** field. The text field to the right of the template code is used as the menu option text. It will populate with the description of the template, but you can override this to something else like "Add Vital Signs" or "New Record". Press **Enter** to add the template to the list.

You can add as many different template options as you want. To reorder them, use the **Move Up** and **Move Down** buttons to the right of the template list. To edit or delete them, click on the template and press **E** or **D** respectively.

You can also let the user select the template based on a category. To do that, select a category in the **Category** field. All templates in that category are displayed in the menu. This option is combined with the template options above - you can use either or both.

## On edit show menu

This section controls what happens when the user selects **Edit** when viewing an EMR list.



On edit show menu

Menu items for open EMR:

☑ &Edit Detail
☐ Preview &Document
☐ Edit &Setup
☑ Sign &Off / Complete
☐ Show Amendment &History

Menu items for signed EMR:

☑ View D&etail
☐ View &Document
☐ View &Setup
☐ Amend / C&orrect
☐ View Amendment &History

There are two sections, corresponding to the **EMR** and **Completed EMR** menu. To disable an option, clear the respective checkbox. You can also customize the descriptions for menu items by editing the text field next to the checkbox.

**Note:** if only one item is selected as in the above image, the menu is skipped and that option is executed automatically.

# EMR List Index

## Overview

The Index refers to how the system will select matching EMR records, and knowing which index to use requires knowing what you want your list to do. Using the wrong type of index can make displaying the list very slow.

Following are examples of some common lists. Each of these would probably use a different index.

- All documents in the EXAM category for a single patient
- Open records for a particular provider
- Records assigned to the current user for signoff
- Prescription records on a certain date
- "Detail" records linked to a "header" record through a custom index

## Index Types

The following index types are available:

- **Patient** is the default option. This should be used if the list will show only one person's data at a time (for example all VITAL records for a specific patient).
- **Encounter Date** is used when you need to show records based on a date (for example all prescription records for today) and not limit them for a specific provider or patient.
- **Owner** is useful for viewing entries owned by a single user, but for all patients and dates (for example report on a specific user's activity). Note that the fewer different users who are the owners of EMR records in your database, the less helpful this type of index will be.
- **Provider / Doctor** index is used for finding entries linked to a single doctor code, across all patients and dates, for example for provider-centric dashboard screens. If the list is also going to be filtered by date and there are only a few providers in the database, using the Encounter Date index may be just as fast, or faster.

- **Location** is used for finding entries linked to a single location code, across all patients and dates, for example for location-centric dashboard screens. If the list is also going to be filtered by date and there are only a few locations in your database, the Encounter Date index may be just as fast, or faster.

- **Case** is used for finding entries linked to a single case number, across all patients and dates. If the list is also going to be filtered by date and there are only a few cases in your database, the Encounter Date Index may be just as fast, or faster.

- **Template** is used for finding entries using one particular template, across all patients and dates. This index is useful for record types that are added once per patient.

- **Category** is used for finding entries that have templates linked to a single category code, across all patients and dates. In almost every case, it is probably better to use any other index than the Category index.

- **Signed** is useful for viewing entries that have been signed off.

- **CloseDate** is used to find all entries that have been signed off on a certain date, similar to the Signed index. The main difference is that the CloseDate index includes the date when the EMR was signed off. If you only need a list of all EMRs that are signed off, then it is faster to use the Signed filter.

## Order and Limit

Below the index type selection are two additional options that control the record ordering and limit.

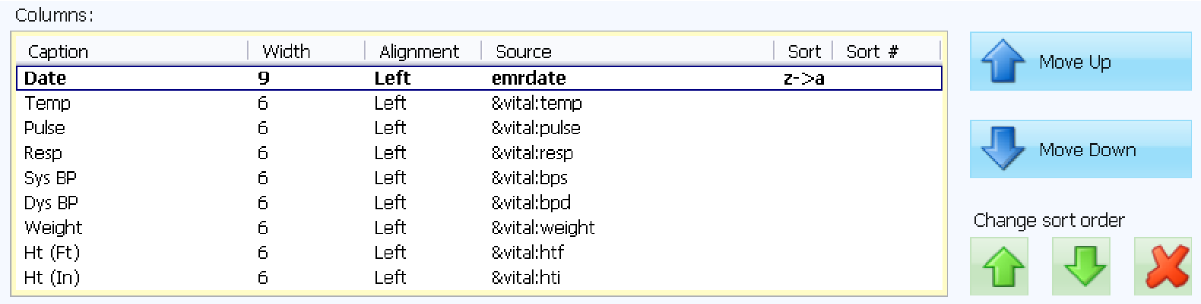
The **Reverse order** checkbox allows you to reverse the sorting order. For example, if you set the index to Encounter Date, the oldest records are shown at the top of the list and the newest records at the bottom. But if you select this option, the order is reversed - newest items are at the top and oldest at the bottom.

The **Limit** field controls how many items are displayed in the list. By default it is set to "all", meaning that all items are displayed. You can specify a number here to limit the list to that number of items (for example to get the last 10 records, specify the index type as Encounter Date and limit the list to 10 items).

# EMR list columns

## Overview

The columns define what data is displayed in your EMR list. You can customize the columns for each list tab.



Columns:

| Caption | Width | Alignment | Source | Sort | Sort # |
|---|---|---|---|---|---|
| **Date** | **9** | **Left** | **emrdate** | **z->a** | |
| Temp | 6 | Left | &vital:temp | | |
| Pulse | 6 | Left | &vital:pulse | | |
| Resp | 6 | Left | &vital:resp | | |
| Sys BP | 6 | Left | &vital:bps | | |
| Dys BP | 6 | Left | &vital:bpd | | |
| Weight | 6 | Left | &vital:weight | | |
| Ht (Ft) | 6 | Left | &vital:htf | | |
| Ht (In) | 6 | Left | &vital:hti | | |

To add a column:

1. Press **A** or right-click and select **Add**. The **Select Type** menu opens.
2. Select the **variable** type.
3. Define the variable(s). Each variable type is described in detail below.
4. Press **F10** to save the column.

The column width is set up automatically to fit the column title and the field values.

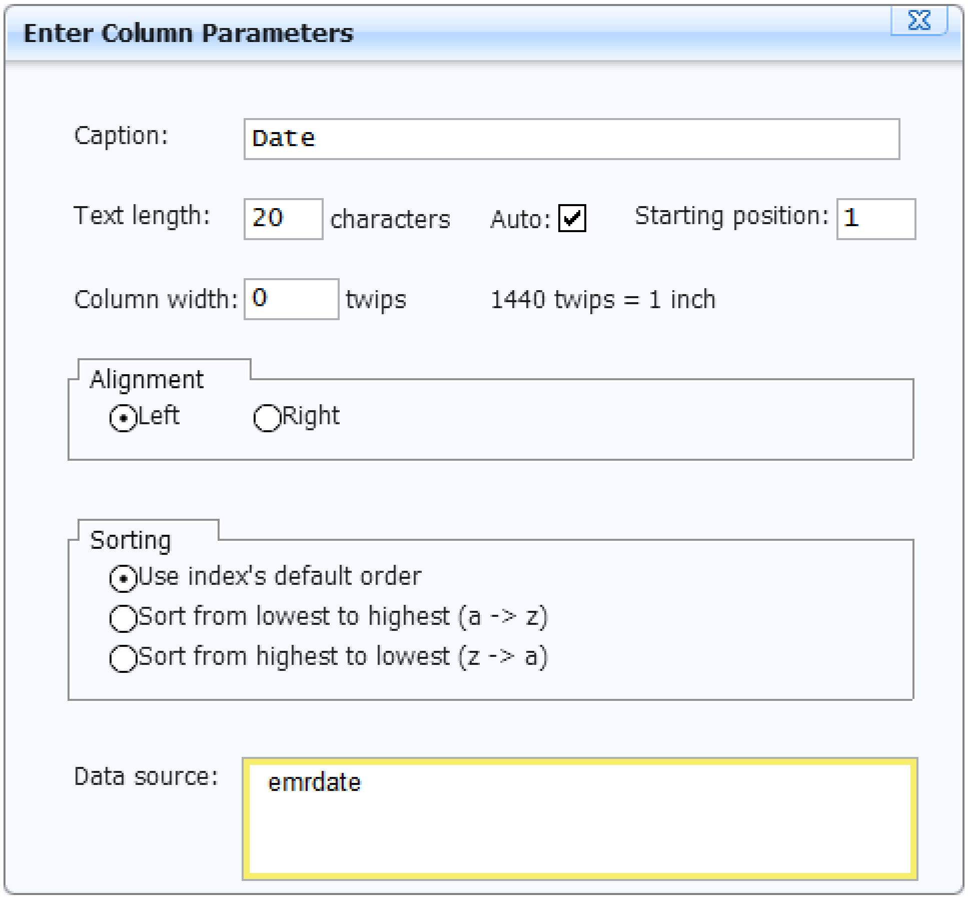
To rearrange the column order, use the **Move Up** and **Move Down** buttons next to the list box. You can delete a column by selecting a column and pressing **D** or right-clicking and selecting **Delete**.

## Editing columns

After you have defined a column, you may want to edit it (change the column alignment or width or edit the column title). For example, you can change the default text length for patient numbers from 11 to 15 characters.

- Double-click on a column definition to edit the column.

  The **Enter Column Parameters** screen appears.



Enter Column Parameters

Caption: Date

Text length: 20 characters Auto: ☑ Starting position: 1

Column width: 0 twips 1440 twips = 1 inch

Alignment
● Left ○ Right

Sorting
● Use index's default order
○ Sort from lowest to highest (a -> z)
○ Sort from highest to lowest (z -> a)

Data source: emrdate

You can edit the following options:

**Caption**

Title of the column in the EMR list.

**Text length**

Number of characters that are displayed in the column. If set to 0, then all text is displayed, in which case the width of the column should be defined using the **Column width ... twips** option.

**Auto**

Resizes the column width automatically.

**Starting position**

Ignores the defined number of characters at the start of a field you are including in the listing. It is set to 1 by default, meaning that no characters are ignored.

For example, you have a field containing `TYPE1 Type 1 Authorization` (the code and description from a table) and you want to ignore the code. In this case you need to change the starting position to 7 , so that the "TYPE1 " part is ignored.

**Column width ... twips**

Fixed width for the column in twentieths of a point. If set to 0 , Raintree uses the **Text length** option.

**Alignment**

Specifies whether the data in the column is aligned left or right.

**Sorting**

Controls how the data in the column is sorted. This allows you to override the sort order specified for the **EMR list index**.

**Data source**

Source for the data in this column.

You can sort the data by multiple columns - first by column 3, then by column 1 and so on. The sort order is displayed in the **Sort #** column. You can change the sort order using the arrow buttons to the right. Clicking the button with a red X removes sorting from the selected column.

## Variables

There are three types of variables available:

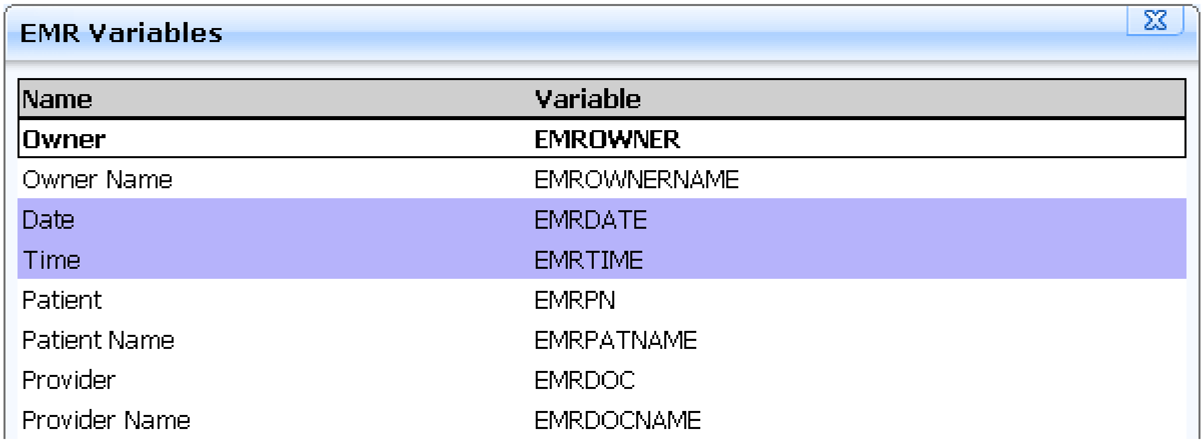
- **EMR Variables** - built-in variables that are available for every EMR record
- **Form Variables** - fields from a custom screen
- **Custom Variables** - custom variables that need to be defined manually

The selection/definition workflow is different for each one of them.

## EMR Variables

This option allows you to use a built-in variable as the **EMR List** column source. These variables (sometimes also called "control fields') are used for tracking the details of an EMR document, such as the patient or when the document was created. Not all of the fields are visible in the built-in EMR record.

When you select this option, the list of all available EMR variables is displayed.



| Name | Variable |
|---|---|
| Owner | EMROWNER |
| Owner Name | EMROWNERNAME |
| Date | EMRDATE |
| Time | EMRTIME |
| Patient | EMRPN |
| Patient Name | EMRPATNAME |
| Provider | EMRDOC |
| Provider Name | EMRDOCNAME |

Select the variables you want to include as columns by highlighting them and pressing **Enter**. The selected variables have a blue background. When done, press **F10** to add all selected variables as list columns with their widths set automatically. Note that the variable names are set as column captions.

The following table lists the variables and their descriptions. Note that here are listed only the fields available when defining **EMR list columns**.

All dates are MM-DD-YY. All times are in the 12 hour am/pm format.

| Field | Formscript Variable | Description |
|---|---|---|
| Owner | EMROWNER | User ID of the user to whom the EMR document is "assigned" |
| Owner Name | EMROWNERNAME | Name of the user to whom the EMR is "assigned" |
| Date | EMRDATE | Date when the EMR record was created |
| Time | EMRTIME | Time when the EMR record was created (12 hour am/pm format) |
| Patient | EMRPN | Patient ID of the patient attached to the EMR record. It is not possible to save an EMR without a patient ID. |
| Patient Name | EMRPATNAME | The first and last name of the patient attached to the EMR record. |
| Provider | EMRDOC | The provider code for the service or event documented by the EMR record. |
| Provider Name | EMRDOCNAME | The provider name. |
| Location | EMRLOC | The location code for the service or event documented by the EMR record. |

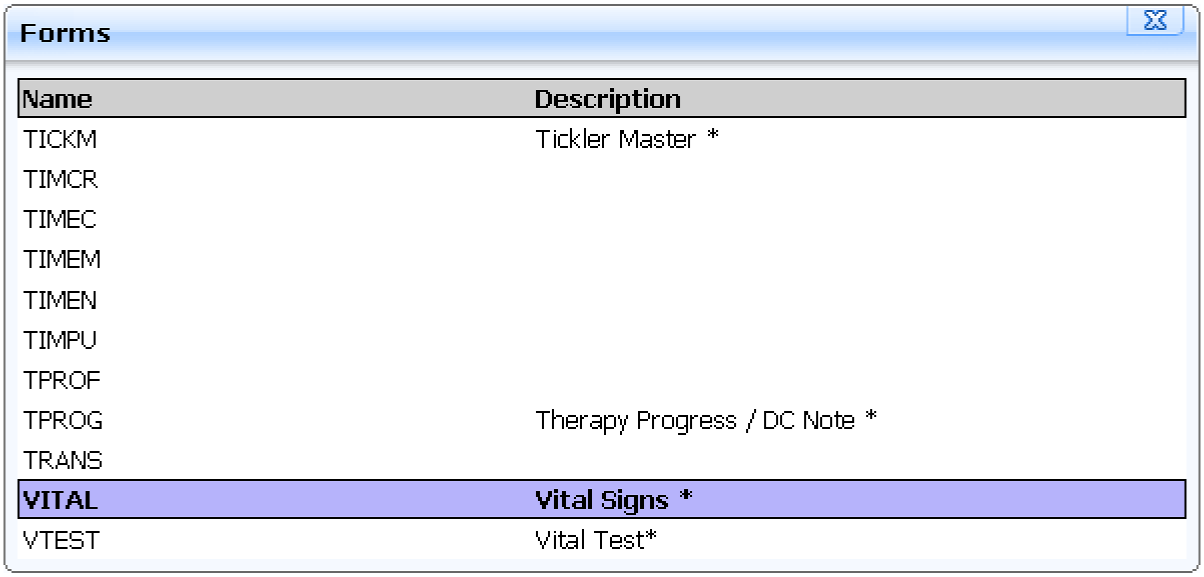
| Field | Formscript Variable | Description |
|---|---|---|
| Location Name | EMRLOCNAME | The location name. |
| Case | EMRPROG | The case number for the service or event documented by the EMR record. |
| Case Name | EMRPROGNAME | The case name. |
| Template | EMRTEMPL | EMR Template used to create the EMR record. While multiple templates can be incorporated into a single record, only one template is the "base" EMR template for each record. |
| Description | EMRDESC | Description of the EMR record. This defaults to the description of the base template, but can be changed through formscript. |
| Category | EMRCAT | The category code of the base template. |
| Category Name | EMRCATNAME | Name of the category. |
| Close Date | EMRCLOSEDATE | Date when the EMR record was signed off |
| Close Time | EMRCLOSETIME | Time when the EMR record was signed off |
| Closed By | EMRCLOSEDBY | User ID of the user who signed off the EMR record |
| Closed By Name | EMRCLOSEDBYNAME | Name of the user who signed off the EMR record |
| Date Created | EMRCREATEDATE | Date when the EMR record was created |
| Time Created | EMRCREATETIME | Time when the EMR record was created |
| Created By | EMRCREATEBY | Used ID of the user who created the EMR record |
| Created By Name | EMRCREATEBYNAME | Name of the user who created the EMR record |
| Date Modified | EMRMODIFDATE | Date when the EMR was last modified |
| Time Modified | EMRMODIFTIME | Time when the EMR was last modified |
| Modified By | EMRMODIFBY | User ID of the user who last modified the EMR record |

| Field | Formscript Variable | Description |
|---|---|---|
| Modified By Name | EMRMODIFBYNAME | Name of the user who last modified the EMR record |
| End Date | EMRENDDATE | A special version of EMRDATE. If EMRDATE is 12-31-79, EMRENDDATE is blank; otherwise it is equal to EMRDATE. |
| Amended | EMRAMENDEDINDICATOR | Indicates whether the EMR has been **amended**. |
| Amendment | EMRISAMENDMENTINDICATOR | Indicates whether the EMR is an **amendment** to a record. |
| Signed | EMRSIGNEDOFFINDICATOR | Indicates whether the EMR has been **signed off**. |
| Duration (Min) | EMRAGEMINUTES | Total time between the EMR creation and the sign off in minutes. |
| Duration (Hrs) | EMRAGEHOURS | Total time between the EMR creation and the sign off in hours. |
| Duration (Days) | EMRAGEDAYS | Time between the EMR creation and the sign off in days. |
| Duration | EMRAGE | Time between the EMR creation and the sign off. If the record has not been signed off yet, EMR shows the time between EMR creation and current time. Note that by default, the age of the last amendment is displayed. To view the ages of individual amendments, view the **Amendment History**. |
| EMR GUID | EMRGUID | Unique identifier for the EMR record. |
| EMR List GUID | EMRLISTGUID | Unique identifier for the EMR list. |

## EMR Form Variables

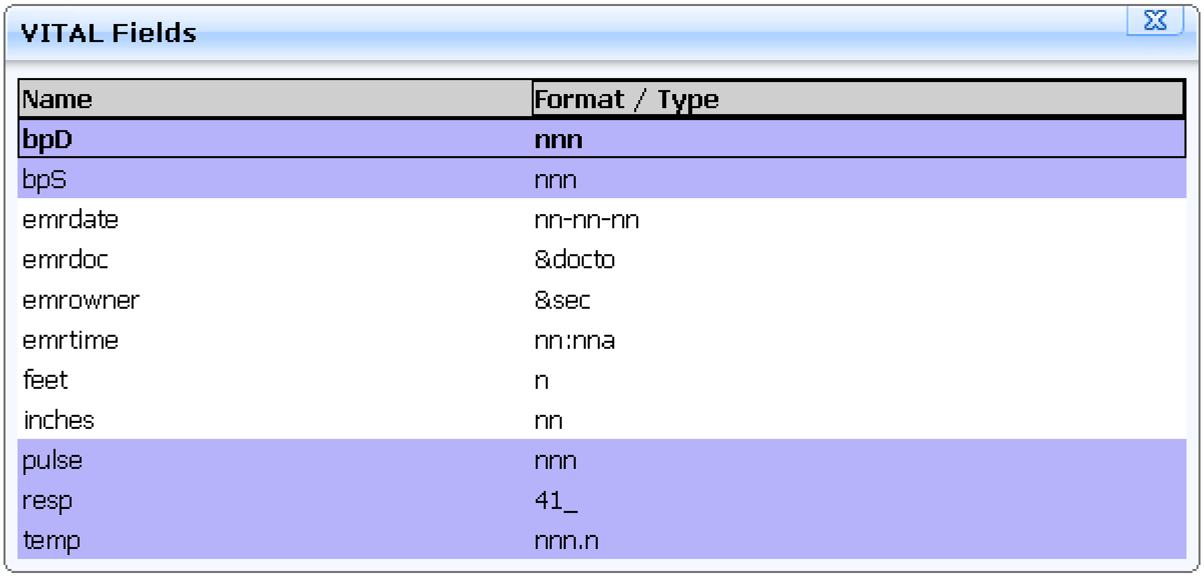
This option allows you to use fields from a custom screen as the **EMR List** column source.

When you select this option, a list of all custom screens is displayed.



| Name | Description |
|---|---|
| TICKM | Tickler Master * |
| TIMCR | |
| TIMEC | |
| TIMEM | |
| TIMEN | |
| TIMPU | |
| TPROF | |
| TPROG | Therapy Progress / DC Note * |
| TRANS | |
| **VITAL** | **Vital Signs *** |
| VTEST | Vital Test* |

Select the screen and press **Enter**. A list of all fields on the screen is displayed.



| Name | Format / Type |
|---|---|
| **bpD** | **nnn** |
| bpS | nnn |
| emrdate | nn-nn-nn |
| emrdoc | &docto |
| emrowner | &sec |
| emrtime | nn:nna |
| feet | n |
| inches | nn |
| pulse | nnn |
| resp | 41_ |
| temp | nnn.n |

Select the fields you want to include as columns by highlighting them and pressing **Enter**. The selected fields have a blue background. When done, press **F10** to return to the list of custom screens.

When you have selected all form variables, press **F10** in the list of screens. All selected fields are added as list columns with their widths set automatically and the field names are set as column captions.

If your custom screen contains any EMR variables, it is recommended to define them directly using the **EMR Variables** option.

### EMR Custom Variables

This option allows you to use a custom variable as the **EMR List** column source. Because it is not possible to determine the column type and width automatically, you have to set up the column options yourself.

When you select this option, the **Enter Column Parameters** screen opens. See the **Editing Columns** section for a description of the fields. The key difference between custom variables and built-in EMR or custom form variables lies in the **Data source** field, where you can define your custom variable.

You can also use formscript expressions as the data source, for example `getdata ("PATIENTREC.FIRST", emrpn)` returns the first name of the patient specified by the EMR patient number. Remember to adjust the column width according to the data returned by the expression.

## EMR List Filters and Conditionals

### Overview

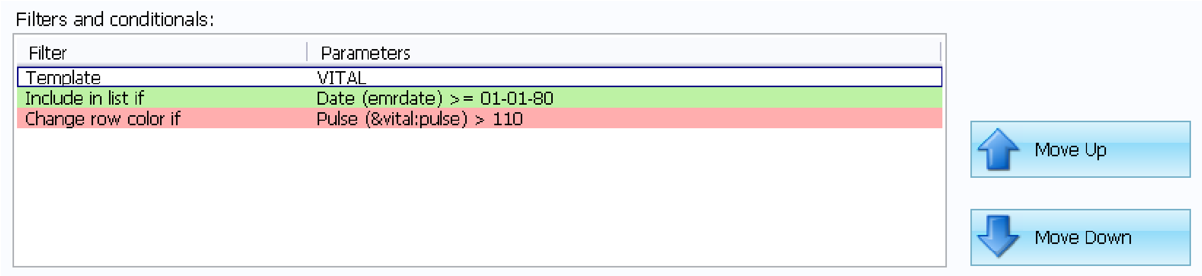
The Filters and Conditionals list has two primary functions:

- limit which records are displayed in the list;
- set up color coding for the rows.

Filters apply only to EMR fields (like filtering for the EMR template type or the provider in the EMR), but conditionals can also filter for custom form fields. While filters simply match values, conditionals can perform other operations as well. You can also use conditionals for color-coding results.

In the image below, there are three filters and conditions specified:

- Only VITAL templates are included in the list.
- Only records created after 01/01/80 are included in the list. Those records are also colored green.
- If the value of the Pulse field is over 110, the row is colored red.



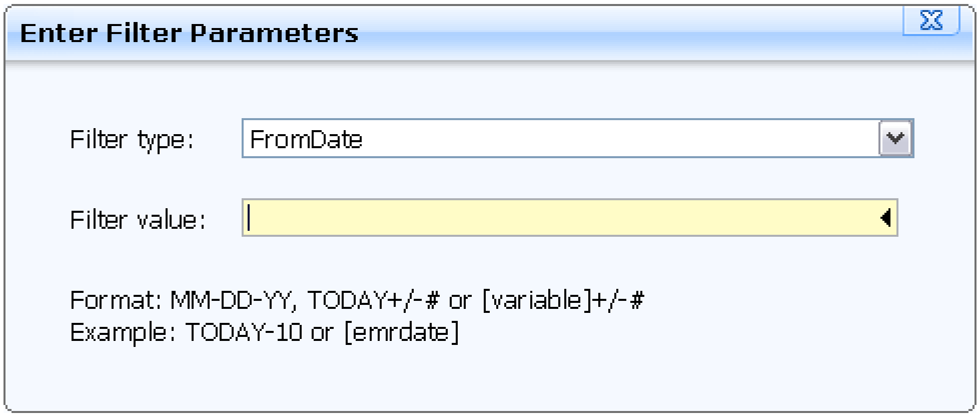
Filters and conditionals:

| Filter | Parameters |
|---|---|
| Template | VITAL |
| Include in list if | Date (emrdate) >= 01-01-80 |
| Change row color if | Pulse (&vital:pulse) > 110 |

You can change the filter order by using the Move Up and Move Down buttons next to the list. Note that you can not move a conditional above a filter or vice versa - filters, conditionals and row color groups are all ordered independently, i. e. you can change the order of individual filters, but all filters will always be applied before any conditions.

The list index determines **how** to look for EMR records to include, but the filters determine **which** records are actually included. For example, we may want all of the unsigned progress notes (using the PNOTE template) for a particular patient. Just specifying the Patient index does not limit the results to progress notes, or unsigned ones. We need one filter for the template type (PNOTE) and another filter for the Open/Signed status (open only).

To add a filter, press **A** or right-click and select **Add** in the Filters and Conditionals listbox. The **Enter Filter Parameters** screen opens:



Enter Filter Parameters

Filter type: FromDate

Filter value:

Format: MM-DD-YY, TODAY+/-# or [variable]+/-#
Example: TODAY-10 or [emrdate]

## Using Custom Variables

In all conditionals and most filters you can also use custom variables. The variable needs to be enclosed in square brackets.

This is very common in dashboard screens, where you have an EMR list and some fields to filter the displayed list. For example, if the screen is named DASH and it contains three fields: TPL for the template code, FDATE for the range start date and TDATE for the range end date, you can set up three filters:

- Filter type: Template, value: `[&DASH:TPL]`

- Filter type: FromDate, value: `[&DASH:FDATE]`
- Filter type: ToDate, value: `[&DASH:TDATE]`

Now, if values in those fields are changed, the EMR List having those filters will be refreshed to contain only records that match the specified values. Note that you need to set the refresh properties of those fields correctly. For more information, see **Using an EMR List in a Screen**.

## Filter Types

The following filter types are available:

- **FromDate** and **ToDate** are used to specify a date range. This only applies to the `emrdate` variable. There are several ways to specify a value for these filter.

  First, you can use a literal date in the format MM-DD-YY (for example 10-31-07). This is not very common, as the list will always show records from that date.

  You can also set a filter based on the current date by using the keyword TODAY. For example, for records dated today, you would just type TODAY. If you want records for the last 30 days, you can use TODAY-30.

  And you can also use a custom variable. This would be very common in lists that are embedded in a screen. For example, if you have a dashboard screen named DASH that has a "starting date" field named STARTDT, you can enter [&DASH:STARTDT] as the filter value to use the value from that field.

- **FromTime** and **ToTime** provide stricter limits for the date range, allowing you to specify a time as well. Using these is similar to the **FromDate**/**ToDate** filters.

- **Patient** is used to show only records for a specific patient. If your list is using the Patient index, you do not need the Patient filter.

  You can specify a patient number as the value, but unless it is a special account (anonymous clients, retail sales, etc.) filtering the list for a specific patient number is not useful. More commonly, the patient number is specified in a field and then used as a variable (similar to FromDate and ToDate). For example, a provider's dashboard may have a patient number field (named PATIENT) to view records related to a specific patient. In that case you can use [&DASH:PATIENT] to set the filter to the value of that field.

- **Owner** restricts the list to records owned by a particular user. The value is often specified as [USERID], limiting the records to ones owned by the user who is currently logged in. Naturally you can also enter a specific user ID as the value here.

  It is also possible to use the Owner filter with a custom variable. An example would be filtering the records on a dashboard screen for a user that you supervise.

- **Provider/Doctor** filter limits the list to records linked to a specific provider code. Like other filter, you can also use either a literal value or a custom variable here (for example to see the records for a supervised provider).

- **Location** restricts the list to records with a specific location code. You can enter a location code (for example if you have a specific walk-in facility or crisis center, you can create a list to view records for that specific location) or use a custom variable.

- **Case** restricts the list to records with a matching case number. As with the other filters, you can use either a specific case number (for example to view all records related to occupational therapy cases) or a custom variable.

- **Template** is used to show only one type of templates in the list. This is a very common filter. If your list needs to show only one type of records, specify the template code here. Naturally, you can also use a custom variable here.

- **Category** limits the list to templates in the specified category. Note that the category is determined at the time the EMR record is created; if you change a template's category, any existing EMR records for that template will still be under the original category.

- **Open/Signed** filter is used if you want to include only unsigned records, or only signed records, but not both. Note that this filter cannot be set with a custom variable.

- **Amendments** controls whether to show only the first amendment (the original version of the EMR), the last amendment or all amendments. By default, the last amendment is shown.

- **Non-Duplicate** limits the list to unique records. The value you enter here indicates what value must be unique.

## Conditionals

Conditionals are more flexible than filters. Instead of simple matching, you can use different operators for comparing the variables and values. Also, conditionals are not limited to built-in EMR variables - you can define a conditional for a field on the screen.

There are three conditionals available. They differ only in the way they are combined with the previous conditionals:

- **AND Expression** requires that the record matches both this and the previously defined conditional. For example, you might want to see records that have a high pulse rate AND a high blood pressure.

- **OR Expression** requires that the record matches either this or the previously defined conditional. For example, you might want to see only records with an abnormal pulse rate (either too high or too low). In that case, you should define two conditionals - pulse greater than 110 OR pulse below 60.

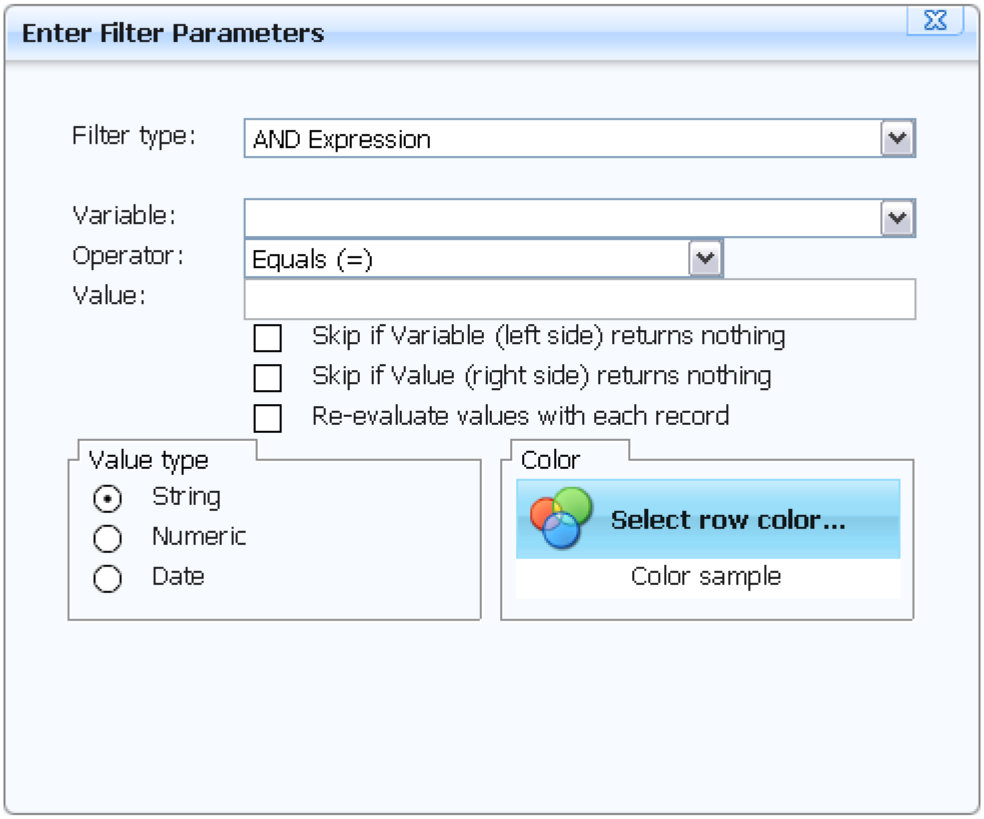
- **Change Row Color** only affects the row color. You can also change row colors with the other conditionals.

> The row colors have to be defined in the correct order, because the first matching condition for row color is applied to the record. This also applies to the colors defined with AND/OR conditionals.

Note that you can also use the conditionals to exclude specific records from the list. For example, you want to see all records except those with the provider code "ANON". In that case you should define an AND conditional - `emrdoc` does not equal ANON.

The AND conditionals are grouped together, so that a record has to match every condition in the group. OR conditionals separate groups, so that you can have (Condition1 AND Condition2) OR (Condition3 AND Condition4). The condition order is very important. For example, if you change the order of the last two conditions, the conditions become (Condition1 AND Condition2 AND Condition4) OR Condition3.

To add a conditional, press **A** in the Filters and Conditionals list and select the appropriate type. Additional fields appear on the screen:


Enter Filter Parameters

Filter type: AND Expression

Variable:

Operator: Equals (=)

Value:

□ Skip if Variable (left side) returns nothing

□ Skip if Value (right side) returns nothing

□ Re-evaluate values with each record

Value type

● String

○ Numeric

○ Date

1. Select the **variable**. This will be checked to see if it meets some condition. The dropdown contains all **columns** you have defined and the special **Select...** option that lets you define a field that is not in your column list.

   For example, you might want to exclude records for a specific program, but do not want to display the program code in the EMR list. Since the Program filter does not support excluding, you need to define a conditional with the program code as the variable.

Selecting the **Select...** option will display the same menu that is used when defining **columns**, containing options for EMR variables, form variables and custom variables. Select the variable the same way as you would when defining a column. Note that here you can only select one variable.

2. Select the operator. This describes how the variable is compared to the value. The following operators are available:

- **Equals (=)** - true if the variable is the same as the value
- **Does not equal (<>)** - true if the variable is not the same as the value
- **Greater than (>)** - true if the variable is greater than the value
- **Less than (<)** - true if the variable is less than the value
- **Greater or equal than (>=)** - true if the variable is greater than or equal to the value
- **Less or equal than (<=)** - true if the variable is less than or equal to the value
- **Contains** - true if the variable contains the value
- **Does not contain** - true if the variable does not contain the value
- **Contained in** - true if the value contains the variable
- **Not contained in** - true if the value does not contain the variable
- **Starts with** - true if the variable starts with the value
- **Does not start with** - true if the variable does not start with the value

3. Enter the value. This can be a literal value like "MED50", a system variable or a custom variable enclosed in square brackets, just like in a filter.

4. The **Skip if Variable (left side) returns nothing** option applies only if the variable is defined as a custom variable. If the variable returns nothing (for example, if the EMR does not contain that field), the condition is skipped.

5. The **Skip if Value (right side) returns nothing** option applies only if the condition value is defined as a custom variable. If the variable returns nothing (for example, if the EMR does not contain that field), the condition is skipped.

6. If the **Re-evaluate values with each record** option is selected, Raintree will evaluate both the variable and the value based on the same record. This way you can apply filters to individual records. For example, you can color all rows where the EMR was created by the provider to whom it was assigned by setting up the conditional so that `emrowner` equals `emrdoc` and then selecting this option.

7. Select the comparison type. Raintree attempts to determine it automatically based on the variable you selected, but you need to verify the selection. An invalid comparison type may result in false matches, so you need to ensure that the correct type is selected.

   The table below describes the results when using different comparison types. **Bold** results are the preferred comparison type. Incorrect results are formatted in *italic*.

| Example | String | Numeric | Date |
|---|---|---|---|
| 01-01-08 = 01-01-08 | True | True | **True** |
| 01-01-08 <> 01-01-08 | False | False | **False** |
| 01-01-08 > 12-31-07 | *False* | *False* | **True** |
| 01-01-08 <= 12-31-07 | *True* | *True* | **False** |
| 1 = 1 | True | **True** | True |
| 1 <> 1 | False | **False** | False |
| 1 < 5 | True | **True** | True |
| 1 > 5 | False | **False** | False |
| 10 <= 5 | *True* | **False** | *True* |
| 10 >= 5 | *False* | **True** | *False* |
| APPLE < PEAR | **True** | *False* | *False* |
| APPLE Starts with "A" | **True** | True | True |
| APPLE Starts with "a" | **True** | True | True |
| APPLE Contains PEAR | **False** | False | False |
| APPLE Contains "P" | **True** | True | True |
| APPLE Is contained in PEAR | **False** | False | False |
| APPLE Is not contained in PEAR | **True** | True | True |

8. (Optional) Select a row color. To do that, click the **Select Row Color** button. The **Select color** screen opens with the available text and background color

combinations. Select the desired color and press **Enter**. The selected color is now displayed below the **Select Row Color** button.

9. Press **F10** to save the conditional.

# Using &SELF in EMR Lists

The special variable `&SELF` is used for referring to a value on the screen or record that is currently open, for example `&self:field1`. In the context of **EMR Lists**, its purpose is slightly different.

- On the **left** side of the filter operator it refers to the records displayed in the list.
- On the **right** side of the filter operator it depends on the **Re-evaluate values with each record** option. If that option is selected, `&SELF` refers to the records in the list; otherwise it refers to the screen containing the list.

# Using EMR Lists

There are several ways to use the EMR Lists. You can:

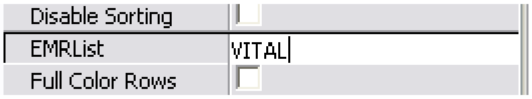
- display the list on a **screen**
- use the list in a **field**
- link the list to a **menu** item
- merge the list into a **narrative**

Each option is described in detail in the respective guide.

## Using an EMR List in a Screen

Once you've created your list definition, you may want to add that list into a screen. This screen could be a dashboard or other custom screen, or it can even be a screen used by an EMR template. To do that:

1. Open the screen.
2. From the **Insert** menu, select **List Box**.
3. Adjust the list box position by dragging it with the mouse. You can also resize it by dragging the edges of the list box.
4. While the list box is selected, the Properties pane on the left should display List Box Widget at the top. Find the EMRList property and enter the list name.



| | |
|---|---|
| Disable Sorting | □ |
| EMRList | VITAL |
| Full Color Rows | □ |

In this example, the list VITAL is linked to the list box.

To add a list from another plugin, use the syntax *plugin*:*list*.lst, for example CHART:visit.lst to add the VISIT list from the CHART plugin.

5. To allow users to add, edit or delete records, you need to set up the **Supported Ops** property. Select the options that you want to allow by checking the respective boxes.

6. (Optional) To display the list as a graph by default, select the Show As Graph property. Note that uses can still toggle between the graph and normal view by using the right-click menu, if Graph is selected under Supported Ops.

7. Press **F10** to save the screen.

## Using a Field to Filter the List

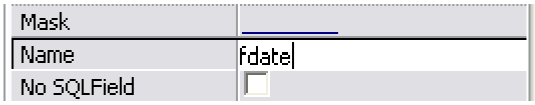
In this example, we want to add a date field to the screen and use this as the From Date filter. The screen is named VTEST and the list we want to show on that screen is named VITAL.

1. Open the VTEST screen.

2. From the **Insert** menu, select **Field**.

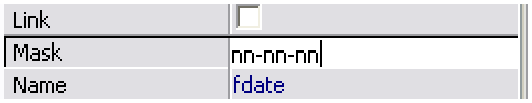
3. Drag the field to the desired position on the screen.

4. Edit the Name property and enter `fdate` as the name.



| | |
|---|---|
| Mask | |
| Name | fdate |
| No SQLField | □ |

5. Edit the Mask property and enter `nn-nn-nn` as the mask.



| | |
|---|---|
| Link | □ |
| Mask | nn-nn-nn |
| Name | fdate |

6. Click on the list box to select it.

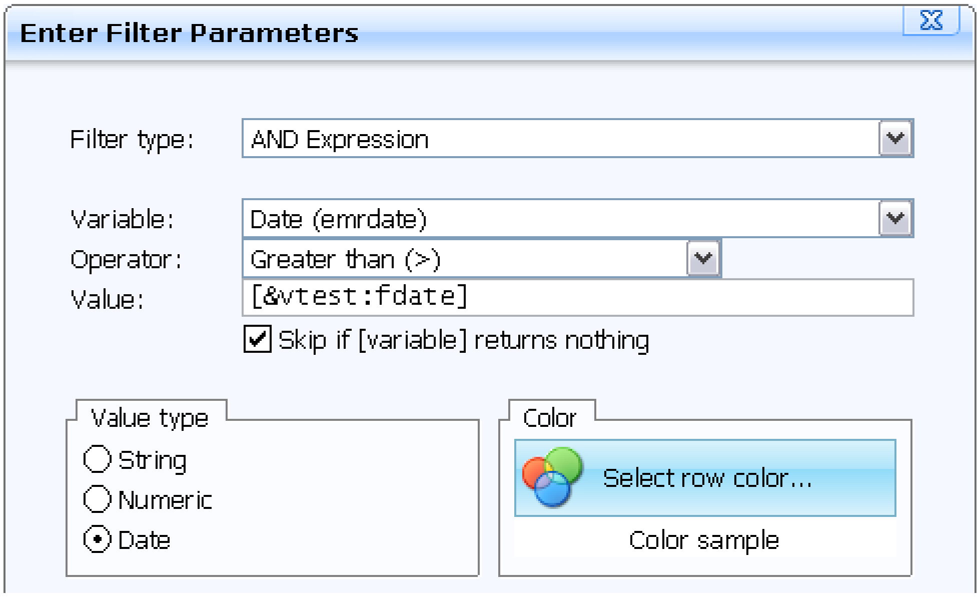
7. Select the **Refresh On Change** property.

8. Change the **Refresh Style** property to `rsOnEnter`.

9. Press **F10** to save the screen.

10. Open the VITAL list in the EMR Lists table.

11. Add a new **AND conditional**. Specify `emrdate` as the variable, `Greater than` as the operator and `[&vtest:fdate]` as the value. Then, select the **Skip if [variable] returns nothing** option and change the comparison type to **Date**.



Enter Filter Parameters

Filter type: AND Expression

Variable: Date (emrdate)

Operator: Greater than (>)

Value: [&vtest:fdate]

☑ Skip if [variable] returns nothing

Value type

○ String
○ Numeric
● Date

This condition means "Only include the record if the EMR was created after the date in that field. Ignore this condition if the field is empty.

12. Press **F10** to save the condition.

13. Press **F10** to save the EMR list.

## Using an EMR List in a Field

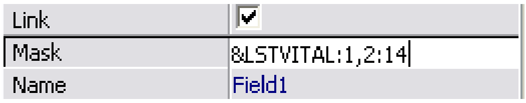
You can also make the EMR list columns available for selection in a field. To do that:

1. Open the screen.

2. From the **Insert** menu, select **Field**.

3. Adjust the field position by dragging it with the mouse. You can also resize it by dragging the right edge of the field.

4. While the field is selected, the Properties pane on the left should display Field Widget at the top. To link the list to that field, edit the Mask property. The value must be in the format &LST`{LIST NAME}:{COLUMNS FROM THE LIST TO INCLUDE}:{WIDTH OF FIELD}`.



| | |
|---|---|
| Link | ☑ |
| Mask | &LSTVITAL:1,2:14 |
| Name | Field1 |

In this example, the first two columns from the VITAL list are displayed and the field width is set to 14.

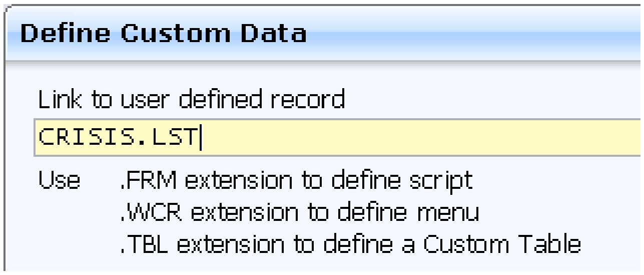
5. Press **F10** to save the screen.

Now the screen has a field linked to the VITAL list. When the user presses **Tab** in that field, the VITAL list is displayed. When the user selects a record from the list (by pressing **Enter**), the field will display the values of the first two columns.

## Using an EMR List from a Menu

Linking an EMR List to a menu allows you to create a simple way to access a list of records, such as records for a certain doctor or location. For example, you may have defined a list of active crisis alerts for all patients, and rather than embedding that list in a screen, you want to link it to a menu option directly from the Reports menu.

1. (Optional) Edit the menu to add a new menu item.
2. Navigate to the menu and press **Tab** on the option you want to use for the EMR list.
3. Type the name of your list, followed by `.LST` in the first field.



Define Custom Data

Link to user defined record
CRISIS.LST

Use .FRM extension to define script
.WCR extension to define menu
.TBL extension to define a Custom Table

In this example, the CRISIS list is linked to the menu.

4. Press **F10** to save.

## Using an EMR list in a narrative

You can also include an EMR list in a narrative. The list will be displayed as a rich text table.

To do that, edit the narrative and include the `<rtftable>` tag in the format `<rtftable:`*`[List name]`*`.lst>`. For example, to include the VITAL list, enter `<rtftable:vital.lst>`.

The `<rtftable>` can also take five additional parameters in addition to the list name:

- `hideempty` will hide any empty rows.
- `fontsize` determines the default font size for the table.
- `font` determines the font for the table.

- `shading:off` will remove the shading from table headers. By default, the table headers have a gray shade as the background, which may cause the headers to disappear when faxing.
- `align` specifies table alignment - left, center or right.

For example,
`<rtftable:VITAL.lst,"hideempty","8","Times New Roman","shading:off", "align:left">` will create a table in 8pt Times New Roman that is aligned to the left.

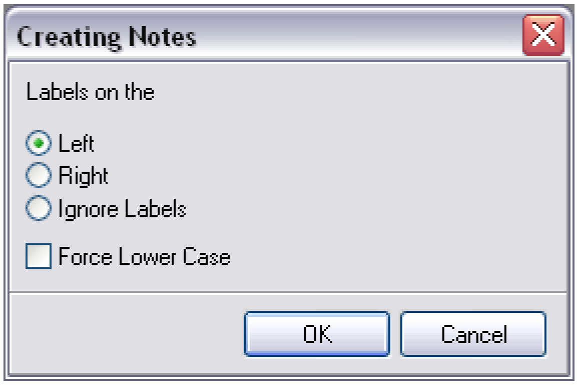
## Template Editor

# Notes

The notes property allows you to specify text that can be inserted into EMR templates for a field depending on its value. The advantage of this approach is that your EMR templates do not have to contain many @if/@endif statements for text that should change depending on the value of a field.

To edit the notes for a widget:

1. Select or add a new widget.
2. Double-click the **Notes** property. You can also click the button with the ellipsis ().
3. For checkboxes and radio buttons an additional window appears, allowing you to specify where the labels for the elements are located.



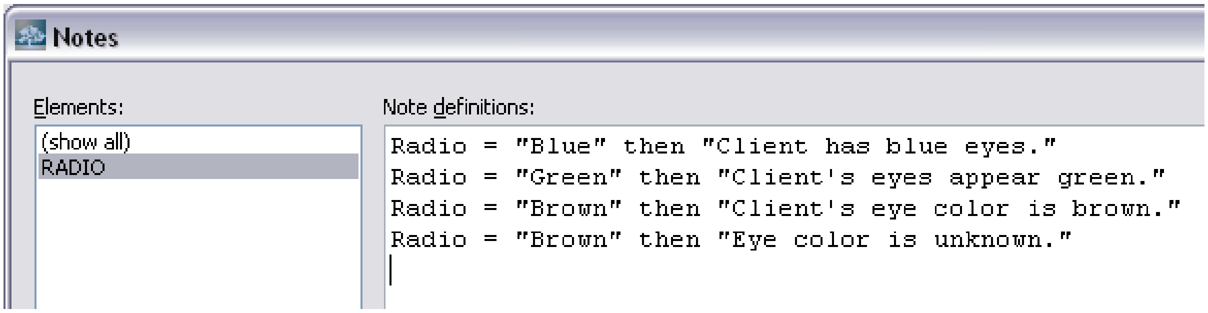
Creating Notes

Labels on the

● Left
○ Right
○ Ignore Labels

□ Force Lower Case

4. The Notes window opens with the current element selected.
5. Specify notes to appear corresponding to different values of the element.



Notes

Elements:
(show all)
RADIO

Note definitions:

```
Radio = "Blue" then "Client has blue eyes."
Radio = "Green" then "Client's eyes appear green."
Radio = "Brown" then "Client's eye color is brown."
Radio = "Brown" then "Eye color is unknown."
```

In this example, different notes are specified for the values of a radio button group. Also, a special note is added for the case when no option is selected.

### Including the value in the note

You can use the variable `%val%` in an item's note. Thus, an example similar to that shown above could have been:

```
Radio <> "" then "Client's eyes are %val%."
Radio = "" then "Client's eye color unknown."
```

You can also compare the value of a field in the notes to insert a note that is context-sensitive, such as a field (in this example TEMP) that recorded a client's temperature. The notes for that field might include:

```
temp > 100 then "High body temperature at %val%."
temp < 97 then "Low body temperature at %val%."
temp ((temp > 97) and (temp < 100)) then "Normal body temperature at %val%."
```

### Using GetData to set the note value

You can set the note value using the **GetData** command. This is useful to replace the actual value with something more meaningful. For example, the stored value of a combobox may be only a code. In this case, you could use `GetData` to set a better value for the note.

The `GetData` command must be surrounded by percent signs. For example:

```
combo_field <> "" then "description is %(GetData("tablename",&SCREEN:combo_field))%"
```

## Note collections and widget groups

Collections allow you to logically group independent fields together, define how the group contents should be displayed, add some text before or after the group, define the order of the group items, control a number of widgets via RSL at once, and so on.

For example, you might want to display a list of symptoms and start the list with "The symptoms include". The symptoms are defined as checkboxes. If you use only regular notes, you have to reference the notes for each checkbox separately and separate the symptoms by commas. But if some of the symptoms are not selected, the list in the EMR will have several commas following each other. Adding the commas in the notes is also not an option, since each symptom might be the last one in the list and therefore does not need a comma. Also, if none of the symptoms appear, you will be left with a sentence "The symptoms include ." One way to solve this would be using multiple conditionals to control the prefix text and commas. The other, much easier way is to define a note collection, defining the list type as AND and specifying "The symptoms include" as the prefix.

Another common use for this feature is scripting. If you define the collection type as Widget Group, you can then handle all widgets in that group at once. For example, you can use this to turn some fields read-only if the user is not allowed to edit them.

To view existing collections in the Template Editor:

1. Select **Template** from the menu bar.
2. Select **Note Collections**.

   The Note Collections window opens.

# Special note collections

Some EMRs can use note collections with a specific name to provide additional functionality. This does not require any scripting - you only need to add the fields to the note collection.

The current special note collections are:

`hide_billed_expected`

Used on: service tickets

This group is used for the billed/expected amounts of charges in the charge selection list. Fields in this group will be hidden unless you have the BILLEXP=E security right.

`billinggroup`

Used on: service tickets

Fields in this group will be read-only after you post the charges.

`cancelledornoshow`

Used on: service tickets

Fields in this note group are set to read-only if the linked appointment is marked as canceled or "No show".

`NoRollForward`

Used on: all EMRs

Widgets in this note group are not rolled forward.

`PullFromCase`

Used on: service tickets

Fields in this group will try to pull their content from identically named fields on the associated case.

`PushToCaseOnSignoff`

Used on: service tickets

When the ticket is signed off, the content of these fields will be written to the associated case.

`pushToCaseAfterSave`

Used on: service tickets

When saving a service ticket, then the content of these fields is copied to the case record; except for the empty values - these are skipped.

`PushToTempl`

Used on: cases

The content of fields in this group will be loaded to the associated service ticket.

**Caution:** using both this collection and `PullFromCase` at the same time may cause problems. You should use this collection unless you have more case types than service tickets.

`stdchg1`, `stdchg2`, etc.

Used on: service tickets

These groups define the posting order of charges. If some charges (`unit_` fields) are in the group `stdchg1`, they are posted before any charges in `stdchg2`, etc.

# Narratives

Narratives are the text documents that accompany EMR records. The narratives usually contain special tags to merge information from the EMR record into the document. Narratives can also contain rich text, so you can use all formatting tools in the RTF Editor - customize fonts, alignment, etc. This allows you to quickly set up clean-looking professional documents that contain all necessary information from your EMR records.

Narratives are stored as .NAR files in docpaths or in the EMR plugins. The filename usually matches the respective EMR template code.

There are two ways to edit a narrative:

- From the EMR plugin by opening an EMR template and selecting the **Edit narrative** option.
- In the Template Editor when editing the custom screen linked to an EMR template.

Both options have their advantages. When you open the narrative from the EMR plugin, it opens in the RTF Editor. This allows you to use all formatting tools to work on the layout. In Template Editor the formatting options are very limited, but you can easily insert tags that correspond to fields on the input screen.

**Note:** to edit an EMR template's narrative from within the Template Editor, you must associate the narrative with the template screen.

Once a NAR file is associated with your screen, you can display an editor for that screen by selecting **Editor** or **Editor and Insert Mode** from the EMR menu.

## Creating narratives

You can add new narratives to the Narratives table, if you want use different narratives with one EMR template; or create a narrative that is linked to a specific EMR template in the **EMR Template** screen.

### Adding narratives to the Narratives table

Narratives that are added to the Narratives table can be selected to an EMR template from the respective field on the EMR screen. To create a narrative in the Narratives table:

1. Press **Tab** in the **Narratives** field to open the Narratives table.
2. Press **A** to add a narrative.

   The **Add narrative** screen appears.
3. Enter a narrative code.
4. Press **F10** to save. The **Narrative Editor** opens.
5. Enter the narrative text.
6. Press **F10** to save.

### Adding narratives to EMR templates

EMR template narratives are linked to EMR templates and to input screens. In this example, a new EMR template narrative (VITAL) is created, which includes a linked input screen (the VITAL screen), and prints the data that is entered in the VITAL screen.

1. Open the **EMR plugins**.
2. Press **Alt+T** to open the **Templates** tab.
3. Press **A** to add a new template. The **EMR Template** screen opens.
4. Enter the template definition.
5. Click the **Create Narrative** button.

   The narrative opens in the editor.
6. Enter the narrative text. If you leave the narrative text empty, it will not be created.
7. Press **F10** to save the narrative.

   The editor closes and the **EMR Template** screen is updated, displaying the path to the narrative.

8. Press **F10** to save.

## Associating narratives with screens

If you want to edit a narrative in the Template Editor, you have to associate it with the screen first. If you created the narrative from the **EMR Template** screen, this was done automatically. Otherwise you have to do it yourself.

1. Open the screen in the Template Editor.
2. Open the **EMR** menu.
3. Select **Associated NARs**.
4. Enter the narrative code.

   You can enter either the narrative code (for example `doc`), narrative file name (for example `doc.nar`), or the narrative name with the plugin name (for example `doc:doc.nar`).

   You can enter multiple codes, but each code must be on a separate line.
5. Click **OK**.

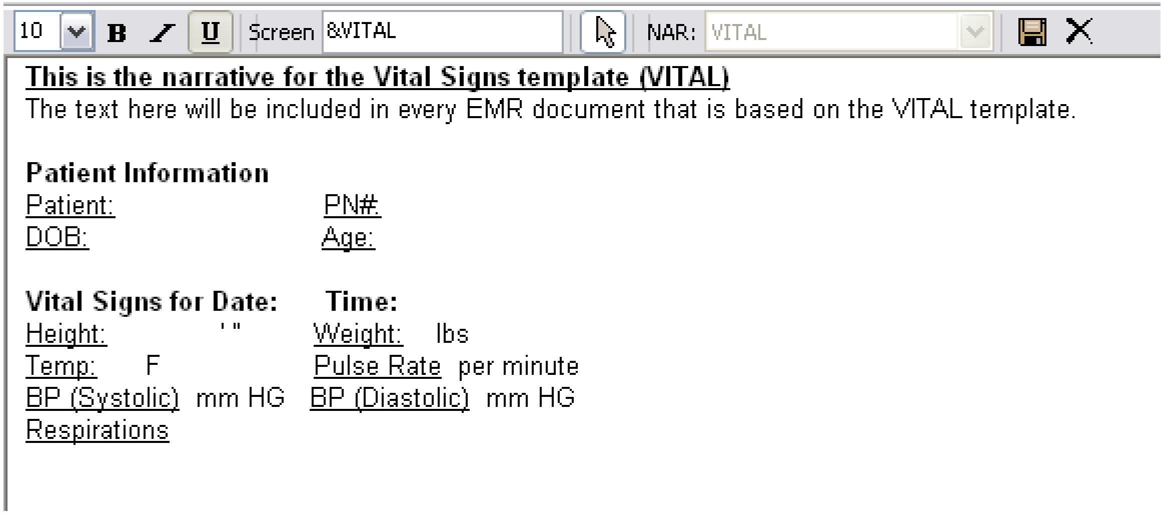
The narratives you specified are now associated with the screen. When you open the Narrative Editor, you can select between the narratives associated with the current screen.

## Adding narrative texts in the Template Editor

This topic gives an overview of how to add narrative texts in the Template Editor. You can add narrative texts in the Template Editor, if you open the associated screen in the Template Editor and select the **Editor** or **Editor and Insert Mode** option in the **EMR** menu on the titlebar.

In the following examples, a new narrative is created for an EMR template (VITAL), which includes a linked input screen (the VITAL screen), and prints the data that is entered into the VITAL screen.

Write the text that you want to have on all EMR documents exactly as you want it to be displayed in the documents.



**This is the narrative for the Vital Signs template (VITAL)**
The text here will be included in every EMR document that is based on the VITAL template.

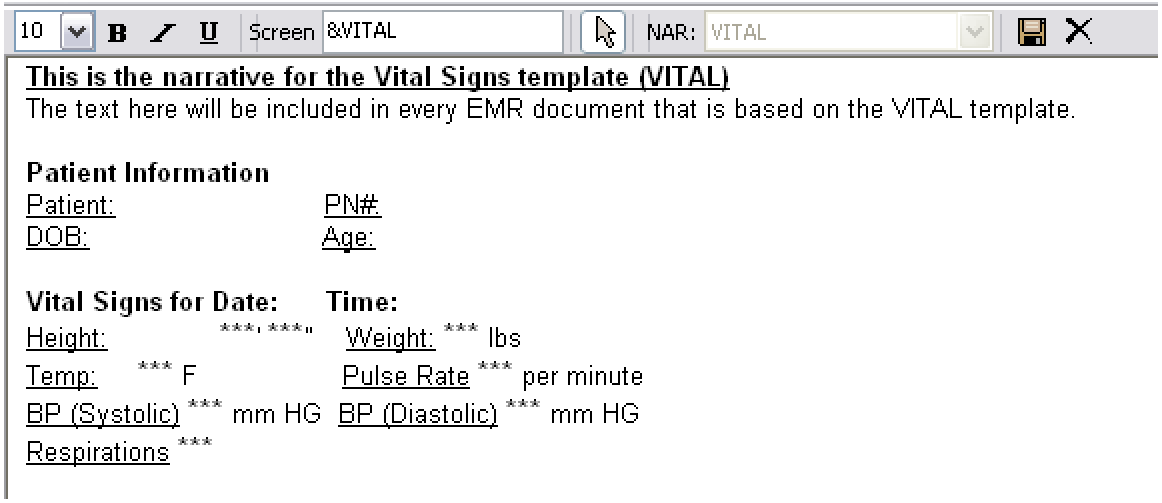
**Patient Information**
Patient: PN#:
DOB: Age:

**Vital Signs for Date: Time:**
Height: ' " Weight: lbs
Temp: F Pulse Rate per minute
BP (Systolic) mm HG BP (Diastolic) mm HG
Respirations

**Tip:** Press **Ctrl+E** to open the narrative in the RTF editor and add any advanced formatting, such as, for example, change the font type and text alignment, or add a bulleted list.

## Inserting data from the screen

You can insert data from the screen into the narrative. For example, in the image below, (*******) mark the places, where data from the VITAL screen will be displayed

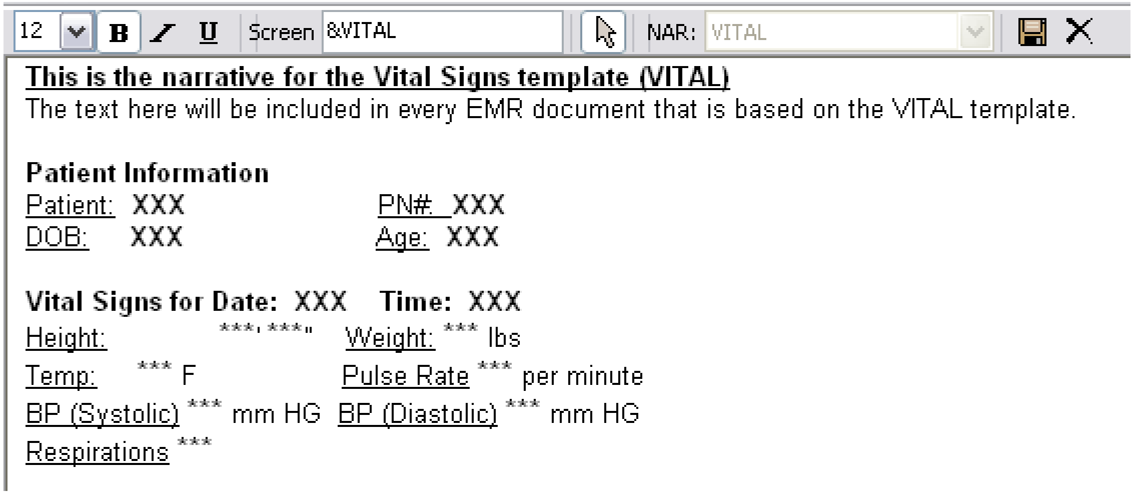


**This is the narrative for the Vital Signs template (VITAL)**
The text here will be included in every EMR document that is based on the VITAL template.

**Patient Information**
Patient: PN#:
DOB: Age:

**Vital Signs for Date: Time:**
Height: ***' ***" Weight: *** lbs
Temp: *** F Pulse Rate *** per minute
BP (Systolic) *** mm HG BP (Diastolic) *** mm HG
Respirations ***

In addition to the fields from the VITAL screen, you can also add data from the EMR records. For example, in the image below, (**XXX**) mark the places, where data from the EMR record will be displayed.



**This is the narrative for the Vital Signs template (VITAL)**
The text here will be included in every EMR document that is based on the VITAL template.

**Patient Information**
Patient: XXX PN#: XXX
DOB: XXX Age: XXX

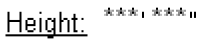
**Vital Signs for Date: XXX Time: XXX**
Height: ***' ***" Weight: *** lbs
Temp: *** F Pulse Rate *** per minute
BP (Systolic) *** mm HG BP (Diastolic) *** mm HG
Respirations ***

## Inserting field values

You can insert field values to narratives by clicking on the field in the template screen. The field value is inserted to the place that your mouse pointer was positioned at.

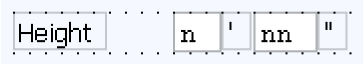
**Note:** You can insert field values ony if you have selected the **Editor and Insert Mode** in the **EMR** menu on the titlebar.

The narrative is displayed in the Narrative Editor as following:



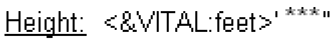
Height: ***' ***"

The fields on the screen are displayed as following:



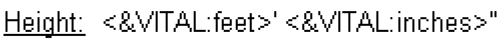
For example, to insert a field for height in feet and inches:

1. Position to the right place in the narrative (before the "'" character).
2. Click on the first **Height** field. The tag `<&VITAL:feet>` is inserted in the insertion point location. This tag tells the EMR system to merge in the value in the **Feet** field of the VITAL screen when the EMR record is saved.

   Height: <&VITAL:feet>' ***"

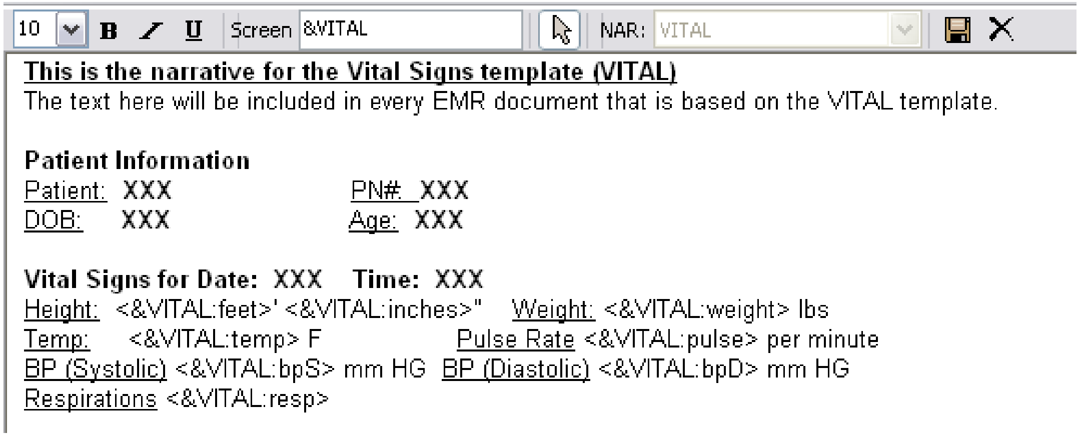
3. Repeat this for the second **Height** field. The narrative changes:

   Height: <&VITAL:feet>' <&VITAL:inches>"

4. (Optional) Repeat the process to add other fields into the narrative.

The end result should look like this:



Screen &VITAL

NAR: VITAL

**This is the narrative for the Vital Signs template (VITAL)**
The text here will be included in every EMR document that is based on the VITAL template.

**Patient Information**
Patient: XXX PN#: XXX
DOB: XXX Age: XXX

**Vital Signs for Date: XXX Time: XXX**
Height: <&VITAL:feet>' <&VITAL:inches>" Weight: <&VITAL:weight> lbs
Temp: <&VITAL:temp> F Pulse Rate <&VITAL:pulse> per minute
BP (Systolic) <&VITAL:bpS> mm HG BP (Diastolic) <&VITAL:bpD> mm HG
Respirations <&VITAL:resp>

## Inserting merge variables

You can also replace **XXX** with the built-in merge variables:

**<EMRPN>**
Patient number.

**<FIRST>** and **<LAST>**
Patient's first and last name.

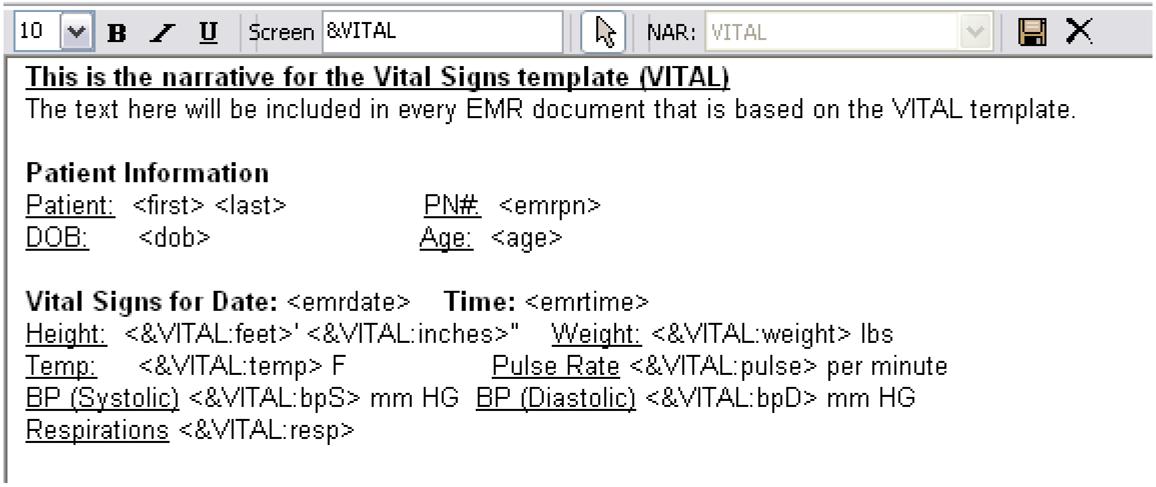
**<DOB>**
Patient's date of birth.

**<AGE>**
Patient's age.

**<EMRDATE>**
EMR date

**<EMRTIME>**
EMR time.

When you have inserted these tags, the narrative should look like this:



**This is the narrative for the Vital Signs template (VITAL)**
The text here will be included in every EMR document that is based on the VITAL template.

**Patient Information**
Patient: <first> <last> PN#. <emrpn>
DOB: <dob> Age: <age>

**Vital Signs for Date:** <emrdate> **Time:** <emrtime>
Height: <&VITAL:feet>' <&VITAL:inches>" Weight: <&VITAL:weight> lbs
Temp: <&VITAL:temp> F Pulse Rate <&VITAL:pulse> per minute
BP (Systolic) <&VITAL:bpS> mm HG BP (Diastolic) <&VITAL:bpD> mm HG
Respirations <&VITAL:resp>

# EMR-related merge commands

## Overview

In addition to all standard **merge tags and commands**, there are special commands and tags that can be useful when creating narratives for EMR templates. The tables below contain the special commands, along with some more commonly used standard merge commands.

## Commands

| Command | Description |
|---|---|
| @if<br>@elseif<br>@endif | These commands can be used for conditional text inside the template. You can only use simple expressions here. Also, each statement must appear on a separate line. Note that if you want to include some text only if a variable is not empty, you should use @block instead. Example:<br><br>`@if <&vital:pulse> > 100`<br>`High pulse rate`<br>`@elseif <&vital:pulse> < 60`<br>`Low pulse rate`<br>`@endif`<br><br>**Note:** You can use the @elseif command without an expression. For example:<br><br>`@if <&vital:pulse> > 100`<br>`High pulse rate`<br>`@elseif`<br>`Pulse rate 100 bpm or lower`<br>`@endif` |
| @block<br>@elseblock<br>@endblock | These commands are used to display data if a data tag (like <DOB>) is not empty. This is useful for conditionally inserting whole paragraphs or sections.<br><br>`@block`<br>`The patient's birthday is <DOB>`<br>`@endblock`<br><br>In this example, if the patient's birthday is empty, the text between the @block/@endblock tags will not be displayed.<br><br>**Note:** If the only tags that are merged between the @block/@endblock command tags are <he>, <him>, <his>, <first> and/or <last>, then the content will also not be displayed. |

| Command | Description |
|---|---|
| @tableblock<br>@endtableblock | These commands are used with RTF tables. If a table is placed between these commands, the result will be the same as placing each row between @block/@endblock commands, meaning that empty rows are not included in the table.<br>`@tableblock`<br>`<rtftable:VITAL.lst>`<br>`@endtableblock`<br>If a row in the VITAL list is empty (meaning there are blank values in all **displayed** columns), that row will not be displayed in the table. |
| forceblock | If this tag is inside a @block, the contents of the block will always be merged. |
| view | Indicates that merged text should be sent to a popup window rather than the main EMR document. |
| emrfilter | This filters the EMRs based on the specified values, similar to the **EMRFILTER** command in RSL.<br>`The syntax is <emrfilter:option:value>`, for example `<emrfilter:category:EXAM>`<br>The available options are the same as for the EMRFILTER command in RSL. |
| emrget | This merges and inserts a narrative into the current EMR. You can use this with the `emrfilter` command to merge a narrative for each EMR found by the filters.<br>You can specify a narrative as an optional parameter. That narrative will be merged instead of the documents of the found EMRs.<br>Example:<br>`<emrfilter:templ:"EMRT1">`<br>`<emrget:"TEST:foo.nar">` |
| emrpopup | This allows you to customize the subsequent EMR popup window (displayed with <view>) to open.<br>`<emrpopup:title:caption>` sets the title for the popup.<br>`<emrpopup:scroll:bottom>` scrolls the content of the popup window to the bottom. |
| EMRPrintImage | This puts an image marker inside the EMR narrative. When the EMR is printed, the images referenced by this marker are printed as the last pages of the document. The syntax is<br>`<EMRPrintImage:"&screenname:imagefield">`<br>This way, the images do not interrupt the rest of the document. |
| emrrestoreindex | This retrieves the current EMR from the stack, similar to the **EMRPOP** command in RSL. |
| emrsetindex | This pushes the current EMR to the stack, similar to the **EMRPUSH** command in RSL. |

| Command | Description |
|---|---|
| | When the command is used without parameters, it will push the EMR specified by previous <emrfilter> commands. You can also use the `<emrsetindex:emrguid>` syntax to ignore the filters and push the EMR specified by the GUID. The parameter can either be a string or a reference to a field on the screen in the format `<emrsetindex:<&screen:field>>` |
| etab | This command is similar to `<shortcut>`. It creates a tab that will run the template indicated by the template code when the tab is clicked. The syntax is `<etab:templatecode:caption>` For example, `<etab:VITAL:Vital Signs>` creates a tab "Vital Signs". When you click this tab, the VITAL template is run. |
| filtercategory | Limits the folders visible in the Templates pane in the EMR editor. You can use `<filtercategory:*>` to display all categories or `<filtercategory:folderlist>` to display only templates from the specified categories. For example `<filtercategory:BH,ASMT>` will display only templates in the BH or ASMT categories. |
| include | Inserts the text in the specified file. Syntax: `<include:filename>` |
| notext | Indicates that the EMR Template does not have a narrative. |
| section | This command is used to create sections within the EMR template. Each section must be closed with an `<endsection>` tag. Sections cannot be nested within a document.<br>The syntax is `<section:[*]categorycode:description [:folderfilter]>`, for example `<section:*MY:My Caseload>`<br>• The **categorycode** specifies the code for the section. If you enter an asterisk before the category code, a tab is created at the top of the document. You can click the tab to move te cursor to the start of that section.<br>• The **description** is used as the section title (and if you created a tab, as the tab title).<br>• The **folderfilter** part is optional. This allows you to associate specific EMR template categories with the section. it acts like `<filtercategory>` within the section. |
| shortcut | This command creates a clickable shortcut button on the right side of the EMR Editor. When clicked, it runs the specified template.<br>The syntax is `<shortcut:[*]template:caption>`, for example `<shortcut:VITAL:Vital Signs>`. If you add an asterisk before the template code, the shortcut is run automatically when the template is run. |

# Inserting fields from custom screens

These commands describe how to insert data from the custom screen attached to your template. Note that you can also use the **narrative editor** in the Template Editor - that way you can insert the tags automatically, without having to worry about the correct syntax or field names.

In all commands below, `screenname` refers to the screen containing the field.

Fields

Use <&`screenname:fieldname`> to merge the content of a field.

Example: <&`SAMPL:PRES`> inserts the PRES field from the SAMPL screen.

Field notes

Use <&`screenname:fieldname:note`> to merge the **note** of a field.

Example: <&`SAMPL:PRES:NOTE`>

Rich text

Use <&`screenname:rtfwidget:content`> to insert the content of a RTF memo into the document.

Example: <&`CHNOT:TXGOAL:CONTENT`>

Images

Use <&`screenname:fieldname:image:width:heigth`> to insert an image from the screen.

Example: <&`SAMPL:Img1:image:3000:3000`

Drawings

Use <&`screenname:buttonname:drawing:width:heigth`> to insert a the content of a drawing.

Example: <&`SAMPL:ImgBtn1:drawing:3000:3000`