Strategic Series 7002

Advanced Custom Screens

Version 9.0

All names of companies, products, street addresses, and persons are part of a completely fictitious scenario and are designed solely to document the use of People-Trak. Similarities to real companies, products, addresses, or persons are purely coincidental.

People-Trak is an extensively customizable HR software product. Screen and report samples rendered in this document reflect the default version of People-Trak. These samples may or may not match the screens and reports within your product if customization has been performed.

Technical Difference, Inc.
5256 S. Mission Road #210
Bonsall, CA 92003
(800) 273-3760
(866) 693-4869 (fax)

www.people-trak.com

(Revision 02/01/2014)

Contents

Notes

Lesson 1
Special Topic: New Hire Automation

Contents

This lesson covers the steps required to automate a sample new hire checklist.

Tutorials

Create a Checklist Report: shows how to create a checklist report used for new hires.

Create a Checklist Report Button Trigger: demonstrates how to create a button trigger that runs the checklist report you just created.

Create a Checklist Screen: shows how to create a checklist screen that uses the button trigger you just created.

Create the Checklist Trigger: demonstrates how to create the checklist trigger based on Hire Date and then shows what happens when the trigger is executed by entering a Hire Date for a new employee.

Tutorial 1.1 – Create a Checklist Report

Let’s do the first step in our new hire automation and create a checklist report.

1. Create a new **Adhoc Report** and name it **New Hire Checklist**.

2. Select the **Show Gridlines** check box and set **Spacing** to **Double**.

3. Leave the **Query** tab blank.

4. On the **Report Definition** tab, select four fields in the order listed below, setting the **Row Modifier** for the last three fields to an asterisk “*”:

Full Name Backward	
User History 1 Comment 1	*****
User History 1 Date 1	*****
User History 1 Date 2	*****

5. For the **Full Name Backward** field, clear the **Repeat** check box and set a **Break** in the **Sort Type** column.

6. Use design mode and set the captions as shown below:

User History 1 Comment 1	**Checklist Item**
User History 1 Date 1	**Start Date**
User History 1 Date 2	**End Date**

7. Close design mode, saving your changes. Run the report.

   There is no data in the checklist for any of the employees. However, you can verify that the format is now correct.

8. Close the **Document Viewer**.

9. Close the **Adhoc Reports** screen, saving your changes. Close the **Explorer**.

10. Continue with the next tutorial.

Tutorial 1.2 – Create a Checklist Report Button Trigger

Now, let’s create a button trigger to run our newly created report.

1. Create a button trigger and set the **Trigger Name** and **Button Caption** to **Checklist Report**.

2. Add a **Display Message** action. Set the **Message Text** to the following:

   Use this report to track the new hire actions while they are occurring. When complete, use the report to update the database.

3. Add a **Run Document** action. Set the document to be the **New Hire Checklist** created in the previous tutorial. Set the **Group Type** to be **Current Record**. Set the **Auto-Destination** to **Viewer**.

   This will cause the checklist to be run for the currently active employee, not for every employee.

4. Close the **Button Triggers** screen.

5. Continue with the next tutorial.

Tutorial 1.3 – Create a Checklist Screen

Now you are going to create the screen that will be the new home for your checklist button.

1. Create a new custom screen and name it **New Hire Checklist**.
2. On the **File** menu, select **Append Screen** and append the **User History 1** screen to your custom screen.
3. Delete the row containing the **User History 1 Comment 2** column.
4. Arrange the order of the columns in the grid so that **User History 1 Comment 1** is first, followed by **User History 1 Date 1**, and then **User History 1 Date 2**.
5. For **User History 1 Comment 1**, set the **Caption** to **Checklist Item**.
6. For **User History 1 Date 1**, set the **Caption** to **Start Date**.
7. For **User History 1 Date 2**, set the **Caption** to **End Date**.
8. Use preview mode to verify that the checklist grid looks right. Make any necessary adjustments. Close preview mode, saving your changes.
9. Create another **Custom Screen** and name it **New Status**.
10. On the **File** menu, select **Append Screen** and append the **Status** screen to your custom screen.
11. Scroll to the near-bottom of the list of objects and insert a row just after the **Screen** object with the name **EMP-2D**.
12. On the new blank line, select **Screen** for the **Object Type**. For the **Object Name**, select the **New Hire Checklist** screen we just created. Set the **Caption** to **Checklist**.
13. Insert another blank line just after the new screen. Set the **Object Type** to **Button** and the **Object Name** to the **Checklist Report** button we created in the previous tutorial. The **Caption** is automatically set to **Checklist Report**, which is the caption we defined when we created this Button trigger.
14. Use preview mode to display the screen. Select the new screen button in the **Object** dropdown on the Tool Bar, which will help you locate that button. Align that button under the existing screen buttons. Make any necessary adjustments to button height or width. Close preview mode, saving your changes. Close the **Custom Screens** screen, saving your changes.
15. On the **Tools** menu, select **Workflows** and modify the **Standard - Employee** workflow. Replace the standard **Status** screen with the **New Status** screen just created. This is the screen that contains the **Checklist** sub screen. Close the **Workflows** screen, saving your changes.
16. Open **Donald Stern's** record and confirm that the **New Status** screen is there. Open the **Checklist** screen to make sure it appears.
17. Close **Donald's** record and continue with the next tutorial.

Tutorial 1.4 – Create the Checklist Trigger

In this tutorial, you will create the trigger that will be executed when a new employee is hired and then set off the new hire automation by entering a new employee record.

1. Create a new **Field Change** trigger and name it **Update Checklist**.

2. Set the **Trigger Field** to **Hire Date**. Keep the **Trigger Use** field set to **Process**.

3. On the **Conditions** tab, set the condition as follows:

If New Value is	**Not Equal to**
	leave blank

4. On the **Actions** tab, add the following **Insert Row** action:

Grid Name	**User History 1**
Insert at Row	**+**

5. Add the following **Update Field** action below the **Insert Row** action, overriding the table:

Field Name	**User History 1 Comment 1**
Row Modifier	**+**
New Value Type	**Value**
New Value	**Handbook Issued**

6. Now, do three more sets of the same **Insert Row** and **Update Field** steps using the following **New Value** entries for the update, overriding the table:

   New Hire Orientation
   Background Check
   Security Badge

   The Insert Row and Update Field actions using the Row Modifier of "+" will add each of these values to the end of the existing data in the grid. In this example above, we could have used Row Modifiers of 1-4 and not bothered with any Insert Row actions because we knew it was a new record and we hadn't entered any data in this grid yet. But to be safe, we recommend you do not assume that there will not be any data there yet. Using the Insert Row and Update Field combination will always add to existing data rather than overwrite existing data.

7. Add a **Display Message** action. Set the **Message Text** to the following:

   New hire checklist has been created. Save the record and then print the Checklist Report by clicking the button captioned 'Checklist Report'.

8. Close the **Field Change Triggers** screen.

9. Click the **New Employee** shortcut, and then click **Create** to create a new employee from scratch. Enter your first, middle, and last name. Select the **New Status** screen we just created.

   After all of the work, we are now ready to see our trigger in action. Ready, set, go!

10. Enter a **Hire Date**. Close the message boxes to keep the trigger running and follow all instructions. (Remember to click the **Save** button before running the report.) When it is all complete, close the **Document Viewer**.

    You should have encountered at least one surprise. There are actually two field change triggers associated with the Hire Date field and both were fired. The trigger created in the previous lesson added company property items and the trigger created in this lesson added a checklist. Let’s confirm this.

11. Select the **Checklist** screen and you will see the checklist. Select the **Property** screen you will see that the standard property has been issued.

    We could have created one large trigger, but that would have taken too much time. It was simpler for this tutorial just to let both triggers fire independently.

12. On the **File** menu, select **Delete Employee** and delete this new employee.

13. This concludes this tutorial and lesson.

Lesson 2
Special Topic: Termination Process

Contents

This lesson covers how to automate a sample termination process.

Tutorials

Create the Wizard Screens: shows how to create the wizard screens that will walk the user through the fields that should be updated when an employee is terminated.

Create the Email Templates: demonstrates how to create the email templates that will be used in the wizard.

Create the Button Trigger: shows how to create a button trigger that brings together the wizard screens and email templates you have created. After adding the button trigger to a custom screen, demonstrates how the whole process is automated by just a click of that button.

Tutorial 2.1 – Create the Wizard Screens

In this tutorial, we are going to create the wizard screens that will be used when an employee is terminated.

1. Create a new **Custom Screen** and name it **WIZ-1**. Set the **Screen Type** to **Popup**. Place a **Frame** object, a **Label** object, and a **Field** object on the screen.
2. Set the **Caption** of the **Label** object to the following:

   Enter the termination date for this employee. Note that the termination date may differ from the employee's last day on the job.
3. Set the field object to the **Termination Date** field. Arrange the screen neatly and size the screen accordingly. Save the custom screen.
4. Create another **Custom Screen** and name it **WIZ-2**. Set the **Screen Type** to **Popup**. Place a **Frame** object, a **Label** object, and a **Field** object on the screen.
5. Set the **Caption** of the **Label** object to the following:

   Enter the termination reason for this employee. Use the support table to pick one of the valid reasons.
6. Set the field object to the **Termination Reason** field. Arrange the screen neatly and size the screen accordingly. Save the custom screen.
7. Create another **Custom Screen** and name it **WIZ-3**. Set the **Screen Type** to **Popup**. Place a **Frame** object, a **Label** object, and a **Field** object on the screen.
8. Set the **Caption** of the **Label** object to the following:

   Enter the termination type for this employee. Use the support table to pick one of the valid termination types.
9. Set the field object to the **Termination Type** field. Arrange the screen neatly and size the screen accordingly. Save the custom screen.
10. Create another **Custom Screen** and name it **WIZ-4**. Set the **Screen Type** to **Popup**. Place a **Frame** object, a **Label** object, and a **Field** object on the screen.
11. Set the **Caption** of the **Label** object to the following:

    Select the check box to specify if the employee is eligible to be rehired in the future.
12. Set the field object to the **Eligible for Rehire** field. Arrange the screen neatly and size the screen accordingly.
13. Close the **Custom Screens** screen, saving your changes.
14. Continue with the next tutorial.

Tutorial 2.2 – Create the Email Templates

In this tutorial, we will create two different email templates that are to be used in the wizard.

1. Create an email template and name the template **Payroll Termination Notification**.

2. Place the template in the **Status** category.

3. Edit the template and enter the following text:

   The employee listed below has been terminated. Please set the employee termination date as indicated.

Employee ID:	**~F~** (insert Employee ID)
Employee Name:	**~F~** (insert Full Name)
Termination Date:	**~F~** (insert Termination Date)

4. Close the **Email Template Editor**, saving your changes.

5. Create another email template and name the template **IT Termination Notification**. Place the template in the **Status** category.

6. Edit the template and enter the following text:

   Employee: ~F~ (insert Full Name)**, ID Number: ~F~** (insert Employee ID) **has been terminated. Please take all appropriate actions to disable network access rights, building access rights, and notify me when this is complete.**

   Thank you,

   ~F~ (insert User Name from the System category)

7. Close the **Email Template Editor**, saving your changes.

8. Close the **Email Templates** screen, saving your changes.

9. Create a new **Adhoc Report** and name it **Terminations This Month**. Set a query to find all employees terminated in the current month. (Use the **START-OF-MONTH** and **END-OF-MONTH** date substitution variables.)

10. For each termination, list the employee's **Full Name**, **Termination Date**, **Termination Reason**, **Termination Type**, and **Eligible for Rehire** status. Sort the results by **Termination Date**.

11. Close the **Adhoc Reports** screen, saving your changes. Close the **Explorer**.

12. Continue with the next tutorial.

Tutorial 2.3 – Create the Button Trigger

Now it’s time to create the button trigger that will start the whole termination process and then click it.

It just so happens that People-Trak already has a Terminate button trigger. We want to change that a bit, so this is a good time to demonstrate how to copy an internal trigger and then make it inactive.

1. On the **Tools** menu, select **Button Triggers**.
2. In the **Trigger Class** dropdown on the right side of the Tool Bar, select **Internal**.
3. Highlight the **TerminateEmployee** trigger.
4. On the **File** menu, select **Copy Button Trigger**.

   The internal trigger is copied to a new user trigger named TerminateEmployee and you are taken to the User triggers display.

   Now, let’s change the actions.

5. Click the **Actions** tab.

   We want to keep the Execute Procedure action that is already there. This updates the last day in the position on the Incumbents screen of the Position Control module. Let’s add all of our actions above that existing action.

6. On the Tool Bar, click the **Insert Row** button eleven times so we have some room to add our own eleven actions.
7. Add a **Display Message** action. Set the **Message Text** to the following:

   You are about to start the termination wizard. Please complete each of the wizard screens as they are displayed. When you are finished, a series of emails will be sent to notify the appropriate personnel that this termination has occurred.

8. Add a **Display Screen** action that displays the **WIZ-1** screen.
9. Add a **Display Screen** action that displays the **WIZ-2** screen.
10. Add a **Display Screen** action that displays the **WIZ-3** screen.
11. Add a **Display Screen** action that displays the **WIZ-4** screen.
12. Add a **Display Message** action. Set the **Message Text** to the following:

    You are about to see a list of the company property that has been checked out to this employee. Please check any of the property that does not have a return date and contact the appropriate personnel to make certain that the property is, indeed, returned.

13. Add a **Display Screen** action that displays the **Company Property** screen. (This is one of the standard screens and not one that was made for this example.)

At this point, we want to send emails and run a document that will include our newly updated information. Both actions only report on information that has been saved to the database, so we need to make sure we save the data first.

14. Add an **Execute Tool** action and open the **Detail** for it. In the **Tool Name** field, click the popup button.

    Several Tool Bar functions are listed. Your triggers can be very versatile using these tools.

15. Select **Save**.

    Now, we can send our emails with the current data.

16. Add a **Send Email** action. Open the **Detail** for that action and set the fields as shown below:

To Address Type	**Position**
To	***use popup to select 1043 - Human Resource Manager***
From Address Type	**Address Book**
From	***use Address Book button to select HR User Timothy Carpenter***
Subject	**Termination Notification**
Template	**Payroll Termination Notification**

17. Add a **Send Email** action. Open the **Detail** for that action and set the fields as shown below:

To Address Type	**Position**
To	***use popup to select*** **1005 - IT Manager**
From Address Type	**Address Book**
From	***use Address Book button to select HR User Timothy Carpenter***
Subject	**Termination Notification**
Template	**IT Termination Notification**

Now, let’s add our action to run the document.

18. Add a **Run Document** action. Add the report that we created in the last tutorial named **Terminations This Month**. In the **Group Type** field, select **Use Query**. Set the **Auto-Destination** to **Viewer**.

    Since that is the last action we want to add, let’s set the internal TerminateEmployee trigger to inactive before we leave this process.

19. In the **Trigger Class** dropdown on the right side of the Tool Bar, select **Internal**.

    The internal Button triggers are listed on the left.

20. Select the **TerminateEmployee** trigger.

21. Click the **Details** tab. Clear the **Active** check box to make that trigger inactive.

22. Close the **Button Triggers** screen, saving your changes.

23. Open **Donald Stern's** record. Select the **New Status** screen we just created.

    After all of the work, we are now ready to see our trigger in action.

24. Ready, set, click the **Terminate** button and follow along. (Be sure to save the data on each screen in order to see the data on the report.)

    After five screens, two messages, two emails, a report and another message from the original execute procedure, the wizard will be complete and you will be free to take further action on Donald Stern's record.

    This elaborate trigger is just a sample of what you can do. You are limited only by what you need to accomplish.

    Since Donald is near and dear to our heart in training, we really don't want to terminate him. Let's reinstate him.

25. Clear the fields in the **Termination Details** frame.

26. Close **Donald Stern's** record, saving your changes.

27. This concludes this tutorial and lesson.

Lesson 3
Special Topic: Offenses

Contents

This lesson covers how to use the customization features in People-Trak to track offenses committed by employees. This technique can be used to track anything for which you need history.

Tutorials

Customize User Fields: shows how to customize user fields for the data needed.

Create Custom Screen: demonstrates how to create a custom screen to be used to capture the data.

Enter Data and Generate Report: shows how to enter the data using your custom screen and then generate a report of that data.

Tutorial 3.1 – Customize User Fields

People-Trak provides all the customization needed to track the history of any information you require. To demonstrate this, we will show you how to track offenses committed by employees. The first step is to customize user fields for the data you need to track. We are going to need a grid to track the offense history, so we will customize the User History 2 grid to meet our needs.

1. On the **Administration** menu, select **Screen/Field Customization** and then select the **Standard - Employee** workflow.

   The Standard - Employee workflow is displayed in screen/field customization mode.

2. In the **Screen** dropdown, select **User**.
3. Click the **User History 2** button.

   The User History 2 screen is displayed. We are ready to begin customizing some of the fields on this grid.

4. In the **User History 2 Text 1** field, press **F1**.

   The Field Properties screen is displayed.

5. Change the **Field Name** and **Screen Caption** to **Offense**.
6. In the **Support Table** field, use the popup to select **User Table 1 - Personnel**.

   We will put offense entries in this table in just a bit.

7. Close the **Field Properties** screen, saving your changes.
8. In the **User History 2 Text 2** field, press **F1**.
9. Change the **Field Name** and **Screen Caption** to **Action Taken**.
10. In the **Support Table** field, use the popup to select **User Table 2 - Personnel**.

    We will put action taken entries in this table in just a bit.

11. Close the **Field Properties** screen, saving your changes.
12. In the **User History 2 Date 1** field, press **F1**.
13. Change the **Field Name** and **Screen Caption** to **Offense Date**.
14. Close the **Field Properties** screen, saving your changes.
15. In the **User History 2 Number 1** field, press **F1**.
16. Change the **Field Name** and **Screen Caption** to **Occurrence**.
17. Close the **Field Properties** screen, saving your changes.
18. Close the **Field Customization s**creen, saving your changes.

19. Close **Screen/Field Customization.**

    Let's go put some entries in the user table for offenses.

20. On the **Tools** menu, select **Maintain Tables**.

21. Select the **User** category.

22. Edit **User Table 1 - Personnel**.

23. Delete any existing entries.

24. Add the following entries to the table:

    Improper Dress
    Late Arrival
    Unauthorized Absence
    Unauthorized Computer Use
    Harassment

25. Close the table, saving your changes.

26. Edit **User Table 2 - Personnel**.

27. Delete any existing entries.

28. Add the following entries to the table:

    Verbal Warning
    Written Warning
    Suspension
    Termination

29. Close the table, saving your changes.

30. Close the **Maintain Tables** screen.

31. Continue with the next tutorial.

Tutorial 3.2 – Create Custom Screen

Now, we are ready to create the custom screen used to enter offenses as they occur. To demonstrate a cool feature of People-Trak, we are going to have individual fields for each of the grid entries we need to make. For demonstration purposes, we will have the actual grid that is being updated below these fields to show how that is getting updated. In reality, you wouldn’t need the grid on the screen at all.

1. On the **Tools** menu, select **Custom Screens**.
2. Create a new custom screen and name it **Offenses**.

   First, we will add the individual fields we are going to enter. We won’t include the Occurrence field because that will be calculated and displayed in the grid.

3. Add the following **Field** objects to the screen:

   Offense Date
   Offense
   Action Taken

   We want each new entry of these fields to be inserted at the top of the grid, so we need to set the Row Modifier to New 1. If we had wanted these fields inserted at the bottom of the grid, we would have used New +.

4. In the **Modifier** column for each of the three field objects, enter **New 1**.

   In order to save this information, we are going to use a Submit button. For now, we will just put in a placeholder for that button. We will create the button trigger needed in just a bit.

5. Add a **Button** object.
6. Add a **Grid** object to the screen. In the **Object Name** column, select the **Container** popup and select the **User History 2** grid.

   The fields in that grid are automatically entered for you in the rows below. We only want the fields we have customized to be on this screen.

7. Delete all **Field** rows that do <u>not</u> include one of our four fields (**Offense**, **Action Taken**, **Offense Date** and **Occurrence**).

   We would like to have our Offense Date field as the first field in our grid and the Occurrence number after the Offense field.

8. Click on the **Offense Date** row and then click the **Row Up** button until that field is right below the **Grid** row.
9. Click on the **Occurrence** row and then click the **Row Up** button so that field is right below the **Offense** row.
10. Click **Preview**.

    Make any adjustments that may be needed.

11. Close preview mode.
12. Close the **Custom Screens** screen, saving your changes.

    Time to create the Submit button trigger.

13. On the **Tools** menu, select **Button Triggers**.
14. Create a new button trigger called **Submit**.
15. Click the **Actions** tab.
16. Add an **Execute Tool** action. In the **Detail** for that action, select **Save** in the **Tool Name** field.
17. Close the **Execute Tool** screen, saving your changes.
18. Close the **Button Triggers** screen.
19. On the **Tools** menu, select **Custom Screens**.
20. Select your **Offenses** screen.
21. In the **Object Name** column of the **Button** row, select the button trigger you just created.
22. Close the **Custom Screens** screen, saving your change.

    The custom screen is complete, so we need to add it to a workflow in order to access it.

23. On the **Tools** menu, select **Workflows** and select the **Standard - Employee** workflow.
24. Click the **Screens** tab and add the **Offenses** custom screen as the first screen.
25. Close the **Workflows** screen, saving your changes.

    We need to do one more thing. When we enter an offense for an employee, it would be nice to automatically know what occurrence number that is. Let’s create a field change trigger that will do that for us.

26. On the **Tools** menu, select **Field Change Triggers**.
27. Create a new field change trigger and name it **Calculate Occurrence**.
28. Place this trigger in the **User** category.
29. In the **Trigger Field** field, click the popup button.

    On the right, you should see all of the user fields that you have renamed. These will also be in the All Fields category, but it is a bit easier to select them quickly this way.

30. Select **Offense**.
31. Keep the **Trigger Use** field set to **Process**.
32. On the **Conditions** tab, enter the condition as follows:

    If New Value Is **Not Equal to** ***leave blank***

This means that when you enter something in the Offense field, this trigger will be executed.

33. Click the **Actions** tab.

    In order to calculate the occurrence number, you need a procedure that will look in the Offenses table, figure out how many times that offense has occurred in the past and then add 1 to that number. You would also probably want to only look back a certain amount of time, let’s say 6 months or about 180 days. When you need procedures such as this, you would normally contact your Support Representative, and we would create a custom procedure for you. For training purposes, we have created a custom procedure that will do just what we want.

34. Add an **Execute Procedure** action and open the **Detail** for it.

35. In the **Procedure Type** field, enter **Custom**.

36. In the **Procedure Name** field, enter **CalcOccurrence**.

37. Close the **Execute Procedure** screen, saving your changes.

38. Close the **Field Change Triggers** screen.

    We are now ready to enter some offenses.

39. Continue with the next tutorial.

Tutorial 3.3 – Enter Data and Generate Report

Let's enter some offense data and then create and generate a report of that information.

1. Open any employee record.

   Your custom screen is displayed because it is the first screen in the workflow. The individual fields at the top will be used for data entry.

2. In the **Offense Date** field, enter a date a few days before today.

   After you enter this information, it will be added to the grid below. You will see this update to the grid happen with every field you enter.

3. In the **Offense** field, use the popup to select **Improper Dress**.

   After pressing ENTER, the field change trigger is executed and calculates the occurrence number. Since this is the first occurrence of this offense for this employee, the Occurrence column for that row in the grid is updated with a 1.

4. In the **Action Taken** field, enter **Verbal Warning**.

5. Click the **Submit** button to save these changes.

   The fields with the New 1 modifier are cleared upon save. The information in the grid changes to the normal color (blue in our scheme) because it has been saved. Let's do one more entry.

6. In the **Offense Date** field, enter today's date.

   Our latest entry is placed at the top of the grid above the previous entry. That is because we used New 1 as the row modifier.

7. In the **Offense** field, use the popup to select **Improper Dress**.

   The trigger is again executed and since this is the second occurrence of this offense, the Occurrence column for that row in the grid is updated with a 2.

8. In the **Action Taken** field, enter **Written Warning**.

9. Click the **Submit** button.

   Once again our data is saved and the New 1 fields are cleared.

   Now let's create a report of this information.

10. Close the employee record.

11. On the **Documents** menu, select **Adhoc Reports**.

12. Create a new report and name it **Offense Report**.

13. On the **Report Definition** tab, enter **Full Name**. Then enter the following fields and use "*" for the **Row Mod** for each:

Offense Date
Offense
Occurrence
Action Taken

14. Run the report.

    This is just a quick and dirty report. You can make it as fancy as you want and use the query to determine what information you would like to focus on.

15. Close the **Document Viewer**.

16. Close the **Adhoc Reports** screen.

    As you have seen, it doesn't take much time to quickly customize People-Trak to be able to track and report on key information.

    Let's do some quick cleanup before concluding this lesson.

17. Edit the **Standard - Employee** workflow and remove the **Offenses** screen from the **Screens** tab. Close the **Workflows** screen, saving your change.

18. This concludes this tutorial and lesson.

Lesson 4
Special Topic: Life Events

Contents

This lesson covers creating a Life Events custom screen to demonstrate the use of several of the objects available in custom screens.

Tutorials

Create the Button Triggers: shows how to create the button triggers required for the custom screen.

Build the Custom Screen: demonstrates how to build the Life Events custom screen.

Use the Custom Screen: shows how to use the custom screen you created.

Tutorial 4.1 – Create the Button Triggers

You have learned about all of the objects available to make a custom screen. Now, it's time to put all those objects together and see them in action. We are going to create a Life Events custom screen that would be used by an employee to add dependents. We'll have some fun with it as well.

Our custom screen is going to use button triggers, so let's create those first.

1. On the **Tools** menu, select **Button Triggers**.
2. On the **File** menu, select **New Button Trigger** or on the Tool Bar, click the **New Button Trigger** button.
3. In the **Trigger Name** field, enter **Submit Life Event**.
4. In the **Category** field, select the **Personal** category.
5. In the **Button Caption** field, enter **Submit**.
6. Click the **Actions** tab.
7. Add an **Execute Tool** action. In the **Detail** for that action, select **Save** in the **Tool Name** field.

   You can do all sorts of other actions upon saving the changes, but you will take a closer look at those when learning more about triggers in later training.

   Now, let's add a couple of buttons that will be used to select gender. You could very easily use the dropdown, but we want to demonstrate the use of pictures on buttons and this seemed like a fun way to do it.

8. Create another new button trigger, and enter **Female** for the **Trigger Name** and place it in the **Personal** category. Leave the **Caption** blank because we will use a picture for this button.
9. On the **Actions** tab, add an **Update Field** action. In the **Detail** for that action enter the following:

Field Name	**Dependent Gender**
Row Modifier	**+**
New Value Type	**Value**
New Value	**Female**

   This will update the Gender field in the last row, which will be inserted prior to this, with a value of Female.

10. Now, create another new button trigger and enter **Male** for the **Trigger Name** and place it in the **Personal** category. Leave the caption blank as before. Add an **Update Field** action and make the same entries as the **Female** button, except of course enter **Male** for the **New Value** instead.

    We are also going to have a helpful links area on our screen. For the sake of time, we are only going to create one link, which is to a parenting website.

11. So, create another new button trigger and enter **Parents** for the **Trigger Name** and **Button Caption**. Place it the **Personal** category. Add a **Hyperlink** action. In the **Detail** for that action, enter a **URL** of **www.Parents.com**.

12. Close the **Button Triggers** screen, saving your changes.

    Ok, the button triggers we need are raring to go, so in the next tutorial we will create our custom screen and use these triggers.

13. Continue with the next tutorial.

Tutorial 4.2 – Build the Custom Screen

In this tutorial, we will build our Life Events custom screen, using the triggers we just created and several of the objects we have learned about.

1. On the **Tools** menu, select **Custom Screens**.

2. On the **File** menu, select **New Custom Screen** or on the Tool Bar, click the **New Custom Screen** button.

3. Name the custom screen **Life Events**.

4. For the **Category**, select **Personal**.

   Let's put a background image on our new screen. We have a picture ready for you to use.

5. Click the **Select Form/Image** button. In the **PT9\SAMPLES** directory of your local People-Trak drive, select the file named **LIFE.JPG**.

6. Add a **Heading1** object to your screen. In the **Caption** column, enter **Life Events**.

7. Add an **HTMLabel** object. In the **Caption** column, click the popup button to display the HTML Editor, and enter the following text:

   Use this screen to add dependents. Click the Submit button to submit your changes to the database.

8. Use the options on the Format Tool Bar to format the text however you would like, and then close the **HTML Editor**, saving your changes.

9. Add a **Frame** object.

   Now, let's add the objects we need on the custom screen. The purpose of this screen is to make entering dependent information easy. Dependent information is stored in the Dependents grid. We want the employee to enter this information without using the grid.

10. In the **Object Name** column, click the popup button to display the **Container** screen.

11. Select the **Grid** container object.

    All grids within the Personnel Management module are listed.

12. Select the **Dependents** grid.

    The fields from that grid are defaulted in the rows below. We will see the next time we preview our screen that these fields are not displayed in a grid, but in the frame we just created.

13. In the **Row Modifier** for each of the fields, enter **New +**.

    This will put the entries in these fields in a new bottom row of the Dependents grid.

14. Move the **Gender** row down to the bottom of the fields.

15. Insert a row above that **Gender** row and add a **Frame** with a **Caption** of **Gender Selection**.

16. In the first blank row, add a **Picture** object. In the **Object Name** column, click the popup button and select our **Female** button trigger. In the **Caption** column, click the popup button. In the **PT9\SAMPLES** directory of your local People-Trak drive, select the file named **PETUNIA.JPG**.

17. Add another **Picture** object and select the **Male** button trigger. In the **Caption** column, select the **PETE.JPG** file in the same location where you found **PETUNIA.JPG**.

18. Add a **Frame** object. In the **Caption** column, enter **Helpful Links**.

    Now we can add the link you created. In reality, this could be an extensive list.

19. Add a **Hyperlink** object. In the **Object Name** column, click the popup button and select our **Parent** button trigger.

20. Add a **Line** object.

    This will be used to separate the Submit button from the rest of the screen.

21. Add a **Button** object. In the **Object Name** column, click the popup button and select our **Submit Life Event** button trigger.

22. Open preview mode.

    As you can see, you need to do some rearranging of the objects on the screen.

23. Take a couple minutes to arrange the objects on the screen as you see fit. This is a good time to refresh your memory on how to select and move objects. For the sake of time, don’t worry about trying to get the screen perfect.

24. Close preview mode, saving your changes.

25. Close your custom screen, saving your changes.

    Let’s create a new workflow for your custom screen.

26. Create a new workflow and name it **Employee Self-Service**.

27. On the **Screens** tab, add the **Life Events** screen. Below that, add the **Dependents** screen so we can verify that our updates are done there as well.

    You would normally have several other custom screens and forms as part of an Employee Self-Service workflow. The possibilities are endless.

28. Close the workflow, saving your changes.

    We are done with the prep work. In the next tutorial, we will give our custom screen a spin.

29. Continue with the next tutorial.

Tutorial 4.3 – Use the Custom Screen

We are now ready to see how all of the objects on our custom screen work together.

1. On the **Workflows** menu, select **Employee Self-Service**.

2. Select **Julie Carrington's** record.

   The workflow is displayed with your custom screen as the first screen. The Gender field will show Female, which is the current value for the last person in Julie's dependents grid (the + row, her daughter). Let's imagine that Julie just gave birth to a bouncing baby boy.

3. Enter whatever you like in the **Dependent Information** fields for Julie's son.

4. In the **Gender Selection** frame, click **Pete** for **Male**.

   Your selection updates the Gender field to now be Male.

5. In the **Helpful Links** frame, click the **Parents** hyperlink.

   You are taken to the Parents website.

6. If you are taking this class via WebEx, close the **Parents** tab on your web browser (leaving your WebEx tab still open) and press **ALT + TAB** to return to People-Trak.
   If you are not taking this class via WebEx, close your web browser, and you will be returned to People-Trak.

   So far we have demonstrated "buttons" using pictures and hyperlinks. We have one more button left, and that is the Submit button, which uses a regular button.

7. Click the **Submit** button to save your changes.

   The fields with the New + modifier are cleared upon save. Let's verify that the Dependents grid has been updated.

8. In the **Screen** dropdown, select **Dependents**.

   You should see your entry in the last row of the grid.

9. Close the record group.

   We just scratched the surface of what is available on custom screens. You can see how easy it is to create any custom screen you desire.

10. This concludes this tutorial and lesson.

Lesson 5
Special Topic: More with Triggers

Contents

This lesson covers how to do apply triggers in even more ways.

Tutorials

Updating Dependent Information: shows how to use button triggers to update dependent information.

Using Buttons with Forms: demonstrates how to use buttons with forms on a custom screen.

Updating Derived Fields on a Screen: shows how to use button and field change triggers to update derived fields on a screen.

Tutorial 5.1 – Updating Dependent Information

In the previous lesson, you created a Life Events custom screen that was used to update dependents. In this tutorial, we are going to add more features to that screen.

We will recreate a simpler version of that screen for our demonstration purposes.

1. On the **Tools** menu, select **Custom Screens**. Create a new custom screen. Name the screen **Life Events 2**.

   Now, let's add the objects we need on the custom screen. The purpose of this screen is to make entering dependent information easy. Dependent information is stored in the Dependents grid. We want the employee to enter this information without using the grid. Let's review our quick and easy way of setting that up.

2. Add a **Frame** object. In the **Object Name** column, select the **Container** popup, select the **Grid** container, and select the **Dependents** grid.

   The fields from that grid are defaulted in the rows below. We will see the next time we preview our screen that these fields are not displayed in a grid, but in the frame we just created.

   We also want to include the contact fields for the dependent.

3. Add another **Frame** object. In the **Object Name** column, select the **Container** popup, select the **Frame** container, and select the **Dependent Address** frame.

4. Delete the **Dependent Smoker** field.

5. In the **Row Modifier** for all of the **Fields**, enter **New +**.

   This will put the entries in these fields in a new bottom row of the Dependents grid and in the associated contact screen.

   Ok, now we have our basic screen in place. We still need our Submit button, but we will take care of that in a minute. Let's add something to our screen that you hadn't done before. We are going to add an Autofill button to the screen that will automatically fill in the contact information for the dependent with the contact information of the employee. There is already an internal button trigger that will do this for you.

6. Add a **Button** object to the custom screen. In the **Object Name** field, use the popup to select **Autofill-D**. Change the **Caption** to be just **Autofill**.

7. Open preview mode and move that button next to the contact fields.

8. Close preview mode, saving your changes.

9. Close the **Life Events 2** custom screen, saving your changes.

   The next thing we are going to do is create the button trigger we need for the Submit button.

10. On the **Tools** menu, select **Button Triggers**. Create a new button trigger and name it **Submit Dependent**. In the **Button Caption** field, enter **Submit**.

11. Click the **Actions** tab.

    We want the submit button to save the information we have entered.

12. Add an **Execute Tool** action. In the **Detail** for that action, select **Save** in the **Tool Name** field.

    Now, for something new. We also want this trigger to recalculate the number in the Dependents field. There is an internal procedure to do just that.

13. Add an **Execute Procedure** action and open the **Detail** for that action.

    The Execute Procedure screen is displayed.

14. In the **Procedure Type** field, enter **Internal**.

15. In the **Procedure Name** field, enter **CalculateDependents**.

16. Close the **Execute Procedure** screen, saving your changes.

17. Close the **Button Triggers** screen.

    Let's add our newly created button to our custom screen.

18. Open the **Life Events 2** custom screen. Add a **Button** object and select the **Submit Dependent** button trigger.

19. Go into **Preview** and position your **Submit** button at the bottom of your screen. Close preview, saving your changes.

20. Close the **Custom Screens** screen, saving your changes.

    Let's give our custom screen a whirl. For time's sake, we will just add this screen to the Standard-Employee workflow.

21. On the **Tools** menu, select **Workflows** and select the **Standard-Employee** workflow. Insert the **Life Events 2** custom screen as the first screen. Close the **Workflows** screen, saving your changes.

22. Open any employee.

    The Standard **-** Employee workflow is opened with the Life Events 2 screen displaying. You should see the Autofill button you added.

23. Enter whatever you want for the name fields, and then click the **Autofill** button.

    Voilà! The contact fields from the employee are filled in for you.

24. Fill in any remaining information.

25. Click the **Submit** button to save your changes.

    The fields with the New + modifier are cleared upon save. Now, let's go confirm the updates were done AOK.

26. Select the **Personal** screen and click the **Dependents** button.

    You should see your dependent information as the last row in the Dependents grid.

27. Click in that last row and then click the **Contact** button.

    The contact information you auto-filled is there.

28. Close the **Dependents Address** screen.

29. Count the number of dependents and then close the **Dependents** screen.

30. Verify that the **Dependents** field on the **Personal** screen agrees with your count.

    You have updated dependent information by not only typing in information but using button triggers to aid you in that update.

31. Close the record group.

32. On the **Tools** menu, select **Workflows** and select the **Standard-Employee** workflow. Delete the **Life Events 2** custom screen from the workflow. Close the **Workflows** screen, saving your changes.

33. Continue with the next tutorial.

Tutorial 5.2 – Using Buttons with Forms

In this tutorial, we will demonstrate how to use buttons with forms on custom screens.

We will first create the custom screen.

1. On the **Tools** menu, select **Custom Screens**. Create a new custom screen and name it **Emergency Contacts Form**.

2. Click the **Select Form/Image** button. In the **PT9\ SAMPLES** directory of your local People-Trak drive, select the file named **EMERGENCY CONTACT FORM.PDF**.

   In order to save some time, we will just add some of the required contact fields.

3. Add the following **Field** objects to the screen:

   Contact 1 Name
   Contact 1 Address 1
   Contact 1 City
   Contact 2 Name
   Contact 2 Address 1
   Contact 2 City

   As with the Emergency Contacts screen in People-Trak, we would like to have Autofill buttons for each emergency contact that will fill in the address of the employee. For one we will use the standard button and for the other, we will use a hyperlink.

4. Add a **Button** object to the screen. In the **Object Name** field, select **Autofill-1**, which is an internal button trigger that will autofill the first emergency contact. Change the **Caption** to **Autofill 1**.

5. Add a **Hyperlink** object to the screen. In the **Object Name** field, select **Autofill-2**, which is an internal button trigger that will autofill the second emergency contact. Change the **Caption** to **Autofill 2**.

6. Open preview mode.

7. Move each of the fields to their respective locations on the screen. We will remove the captions in a bit.

8. Move the **Autofill 1** button to the primary contact area of the screen.

9. Move the **Autofill 2** hyperlink to the secondary contact area of the screen.

10. Close preview mode, saving your changes.

    We don’t need our field captions because the form itself has those captions on it.

11. Clear the **Caption** on all **Field** objects.

12. Open preview mode to see that the screen is now looking much better. Close preview mode.

13. Close the **Custom Screens** screen, saving your changes.

14. Edit the **Standard - Employee** workflow and insert the **Emergency Contacts Form** screen after the **Personal** screen.

15. Close the workflow, saving your change.

16. Click the **New Employee** shortcut, and then click **Create** to create a new employee from scratch.

    We will enter just the fields we need for our demonstration.

17. Enter anything you would like for the **First Name**, **Last Name**, **Address 1** and **City** fields. Click the **Save** button to save those changes and stay in the record.

18. Switch to the **Emergency Contacts Form** screen.

19. In the **Name** field for the first emergency contact, enter whatever you would like.

20. Click the **Autofill 1** button.

    The address information from the employee is autofilled for the address of this first emergency contact.

21. In the **Name** field for the second emergency contact, enter whatever you would like.

22. Click the **Autofill 2** hyperlink.

    The address information from the employee is autofilled for the address of the second emergency contact.

23. Now, go back to the **Personal** screen and click the **Emergency** button.

    All of your updates are there.

    This is just more proof that a screen is a screen is a screen. You created a screen that uses a PDF form as the background. You can add fields, buttons, hyperlinks and other objects to your screen, use that screen in a workflow with other standard or custom screens, and have all of the information shared among the screens.

24. Go ahead and delete this new employee.

25. Continue with the next tutorial.

Tutorial 5.3 – Updating Derived Fields on a Screen

There might be times when you want to have a derived field on a screen or form and want to see that derived field automatically updated when you change any of the fields that are part of its formula. Let's take a look at how to do that.

First, we need to create the derived field.

1. On the **Tools** menu, select **Derived Fields**.

   Create a new derived field. Enter **Next Annual Pay** in the **Field Name** and **Caption** fields. In the **Field Type** field, enter **Numeric**. In the **Formula** field, multiply the **Next Rate** field by the **Next Periods** field. The formula should look like the following when you are done:

   ~SAL_NXT_RATE~ * ~SAL_NXT_PER~

2. Close the **Derived Fields** screen, saving your changes.

3. Create a new custom screen and name it **Calc Next Annual Pay**.

4. Add the following fields to the screen and make each **Caption** match the **Object Name** for ease of reference:

   Next Rate
   Next Salary Frequency
   Next Periods
   Next Annual Pay

5. Close the **Custom Screens** screen, saving your changes.

6. Edit the **Standard - Employee** workflow. Replace the **Emergency Contacts** screen with the **Calc Next Annual Pay** screen. Close the workflow, saving your changes.

7. Open any record.

   The Standard - Employee workflow is opened with the Calc Next Annual Pay screen displayed.

8. In the **Next Rate** field, enter **3000**. In the **Next Salary Frequency**, select **Monthly**.

   The Next Periods field is updated because there is an internal field change trigger on the Next Salary Frequency that updates the Next Periods field.

   But our Next Annual Pay field is not updated. This is because it is a derived field that you created, and you need to tell People-Trak that you want that recalculated as well. So, let's do just that.

9. Close the record group, without saving your changes.

10. Create a new **Field Change Trigger** and name it **Pay Rate Change Trigger**. In the **Trigger Field**, enter **Next Rate**. On the **Conditions** tab, enter the following condition:

If New Value is	**Not Equal to** ***leave blank***

11. On the **Actions** tab, enter **Update Derived** as the only action.

    There is no detail for this action because it simply updates all derived fields on the current screen.

12. On the Tool Bar, click the **Save** button.

13. On the **File** menu, select **Copy Field Change Trigger**.

    A new trigger is created as a copy of the Pay Rate Change Trigger.

14. Change the **Trigger Name** to **Salary Frequency Change Trigger**. In the **Trigger** Field, enter **Next Salary Frequency**. On the **Conditions** tab, enter the following condition:

If New Value is	**Not Equal to** ***leave blank***

15. Close the **Field Change Triggers** screen, saving your changes.

    Now, let’s see how our triggers work.

16. Open any employee.

17. In the **Next Rate** field, enter **3000**.

    Since that field changed, the Next Annual Pay derived field is updated. It will stay as 0 if there isn’t a Next Salary Frequency value yet.

18. In the **Next Salary Frequency** field, enter **Monthly**.

    The Next Annual Pay field is updated again. The update of the derived field is being kicked off by field change triggers.

19. Close the record group.

    We could have used a button to update the derived fields on the screen. Let’s try that method instead.

20. Delete the two **Field Change Triggers** you created.

21. Create a **Button Trigger** with a **Trigger Name** and **Button Caption** of **Update Derived**. Add an **Update Derived** action. Close the **Button Triggers** screen, saving your changes.

22. Edit the **Calc Next Annual Pay** custom screen and add a **Button** object that uses the **Update Derived** button trigger you just created.

23. Preview the screen and make adjustments as needed.

24. Close the **Custom Screens** screen, saving your changes.

25. Open any employee.

26. In the **Next Rate** field, enter **3000**. In the **Next Salary Frequency** field, enter **Monthly**.

    The Next Annual Pay was not updated because we removed the field change triggers on those fields.

27. Click the **Update Derived** button.

    That updated our derived field. So, you have the option of using Field Change triggers or Button triggers to update derived fields on the current screen.

28. Close the record group, without saving your changes.

29. Remove the **Calc Next Annual Pay** screen from the **Standard-Employee** workflow.

30. Close the **Workflows** screen, saving your changes.

31. This concludes this tutorial and lesson.