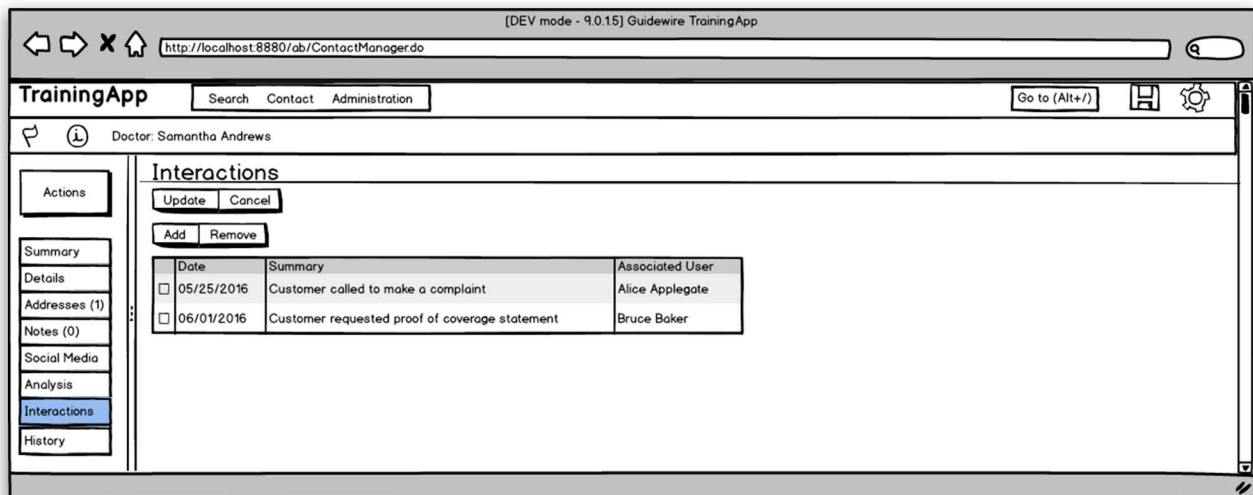


Lesson 2 List Views

The business analysts want to capture specific information about each contact interaction such as a telephone call, email message, postal mail letter, or in-person office visit. Eventually, they want the information displayed in an Interactions section of TrainingApp.



"We want a CSR (Customer Service Representative) such as Alice Applegate to be able to create and edit details about contact interactions." – Insurance company business analysts

In the previous lab, you created the `Interaction_Ext` entity and modified the `ABContact` entity to capture interactions. In this lab, you will implement the necessary **user interface changes** to display the interactions. You will also make the ListView editable in a later lab. As a configuration developer, you will use the PCF Editor in Guidewire Studio to modify the TrainingApp user interface.

2.1 Prerequisites

You must first complete the following previous lesson(s):

- Creating New Entities

For this lab, use TrainingApp, Guidewire Studio, and a supported web browser.

`http://localhost:8880/ab/ContactManager.do` is the default URL for TrainingApp. To view, edit, and delete various contacts, log in to TrainingApp as Alice Applegate. The login/password for Alice Applegate is `aapplegate/gw`.

2.2 Lab: Create a List View

As a configuration developer, you want to be able to create reusable list view panels to be able to display information about multiple entities. In this lab, you will use the PCF Editor to create a reusable List View.

1. Create and configure the InteractionsLV PCF file:

- a) Create a List View Panel in the `traininglabs` folder
- b) Specify a root object of the type `ABContact`
- c) Each row of the List View must display the following:
 - `InteractionDate` field labeled as `Date`
 - `Summary` field labeled as `Summary`
 - `AssociatedUser` field labeled as `User` (use the `AltUserCell` widget)

2. Use Studio's code generation feature to process the PCF file and generate all the resources

- a) If possible, use incremental code generation
- b) Verify that there were no errors during code generation
- c) Verify that you can see generated resources (`.gs`, `.pcf`, `Expressions.gs`)



Hint

Configuring cells

Remember, you can always use the Data Dictionary, the generated Java class, and the widget reference table to find out the name, database type and Gosu type of a field that needs to be displayed. Cells are atomic widgets and similar to input widgets.



Hint

Helpful hints that help you to completion.

The required variables tab defines the input parameters for a container, while the variables tab defines local variables to store values temporarily. (E.g. storing the result of an expensive function call to use it at multiple places in the PCF)

2.3 Lab: Reference the List View

As a configuration developer, you want to be able to reference reusable containers. In this lab, you will add the InteractionsLV to the ABContactInteractionsPage PCF file.

1. Navigate to the ABContactInteractionsPage and replace the inline Detail View

- a) Delete the Detail View
- b) Add a reference to the InteractionsLV
- c) Add a Toolbar to support paging



Hint

Toolbar

Every List View needs to have a Toolbar associated with it. Don't forget that List Views support pagination and the buttons will automatically appear when the number of elements exceeds the pageSize.

2.3.1 Verification



Activity

Verify the work you have done

1. Log in to TrainingApp as Alice Applegate
2. Use the appropriate shortcut to deploy the changes
3. View the Interactions page for William Andy
 - a) Navigate to William Andy
 - b) In the sidebar menu, click Interactions

- c) Verify that you see the Interactions List View (Note: There will be no data in the List View for now. This is because there are no interactions in the database yet. You will be able to insert new interactions in the Editable List Views lesson.)

The screenshot shows a user interface for viewing interactions. On the left is a dark sidebar with a list of menu items: 'Actions' (highlighted in green), 'Summary', 'Details', 'Addresses (3)', 'Notes (5)', 'Social Media', 'Analysis', and 'Interactions' (highlighted in blue). The main content area has a white header with 'Person: William Andy' and a large title 'Interactions'. Below the title is a light gray bar containing a green 'Edit' button, a vertical separator, and a table view icon. Underneath this bar is a table header with three columns: 'Interaction Date', 'Summary', and 'Associated User'. The table body is empty, displaying the text 'No data to display'.

Person: William Andy

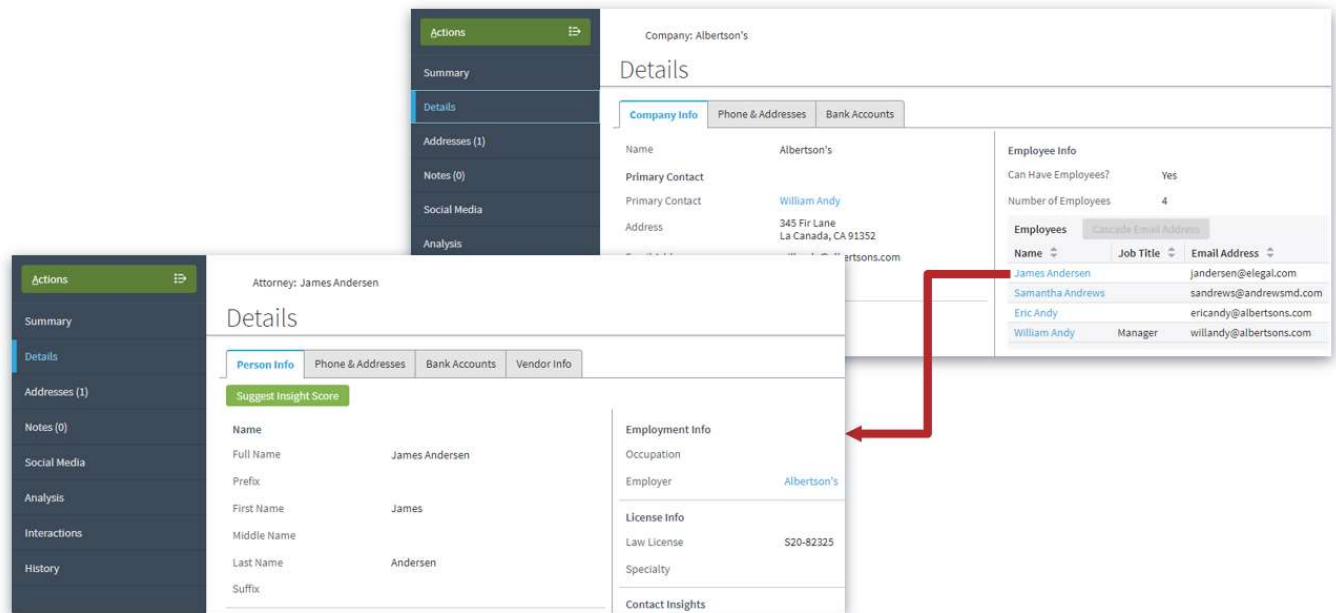
Interactions

Edit

Interaction Date	Summary	Associated User
No data to display		

2.4 Bonus lab: List Views and navigation

“In the screenshot below, the list view panel shows all the employee contacts associated with the employer. Currently, there is no navigable link in the list view panel to the employee contact. A user would need to search for the specific employee contact to see more details. Modify the list view panel so that each employee name links to the contact details page. For example, clicking on William Andy in the table would take us to the Details page of William Andy.” – Insurance company business analysts



Hint

Navigation

If needed, review the **Introduction to Locations** lesson.

2.4.1 Verification

1. Log in to TrainingApp as Alice Applegate
2. Reload the PCF changes

3. Go to the Details page for Albertson's

- a) Confirm that you are able to click a navigable link for each employee contact

Actions

Summary

Details

Addresses (1)

Notes (0)

Social Media

Analysis

Interactions

History

Company: Albertson's

Details

Company Info

Phone & Addresses

Bank Accounts

Name

Albertson's

Primary Contact

Primary Contact

William Andy

Address

345 Fir Lane
La Canada, CA 91352

Email Address

willandy@albertsons.com

Prefers Contact By Email?

No

Additional Info

Tax ID

Employee Info

Can Have Employees?

Yes

Number of Employees

4

Employees

Cascade Email Address

Name	Job Title	Email Address
James Andersen		jandersen@elegal.com
Samantha Andrews		sandrews@andrewsmd.com
Eric Andy		ericandy@albertsons.com
William Andy	Manager	willandy@albertsons.com



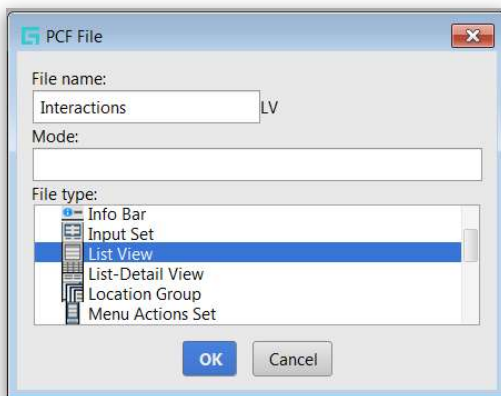
2.5 Lab Solution: Create a List View



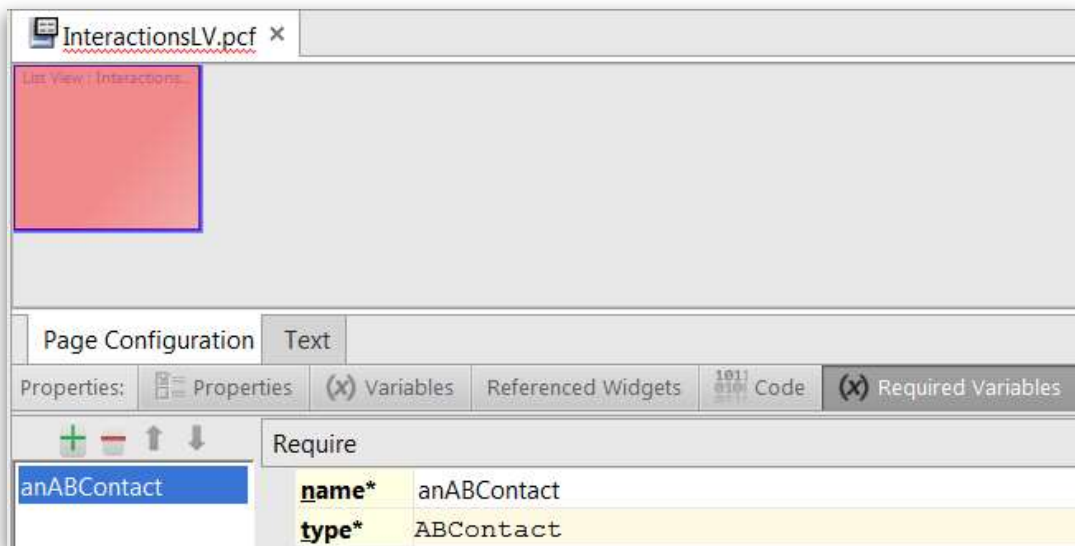
Solution

Exact details on how to complete the lab.

1. Create traininglabs PCF folder if it doesn't exist.
2. Create the List View.



3. Specify the root object.

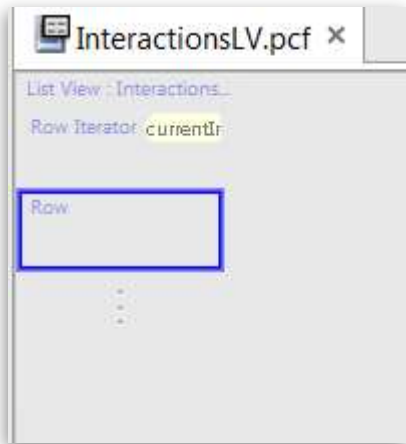


4. Add and configure the Row Iterator

The screenshot shows the SAP NetWeaver IDE interface for a project named 'InteractionsLV.pcf'. The main editor area displays a 'List View : Interactions...' with a 'Row Iter...' component highlighted in a red box. Below the editor, the 'Page Configuration' tab is active, showing a 'Text' view. The 'Properties' tab is selected, displaying the configuration for the 'Row Iterator' component. The 'Basic properties' section is expanded, showing the following properties:

Property	Value
editable*	true
elementName*	currentInteraction
id	
toAdd	
toCreateAndAdd	
toRemove	
value*	anABContact.Interactions_Ext
valueType*	Interaction_Ext[]

5. Add the Row



6. Add the Date Cell

The screenshot shows the 'InteractionsLV.pcf' editor. At the top, the 'List View : Interactions...' is selected. Below it, the 'Row Iterator' is set to 'currentInte'. A 'Date' cell is visible in the list view. A blue box highlights the date format icon (showing '..../....' and a calendar) in the 'Row' section. Below the list view, the 'Page Configuration' tab is active, showing the 'Text' section. The 'Properties' tab is also visible, showing the 'Date Cell' properties.

Date Cell Properties:

Basic properties	
action	
dateFormat	<none selected>
editable	true
id*	InteractionDate
label	DisplayKey.get("Ext.InteractionDate")
required	
timeFormat	<none selected>
value	currentInteraction.InteractionDate

7. Add the Text Cell

The screenshot shows the Xcode interface for editing a table view. The top section displays the table view structure with columns 'Date' and 'Summary'. A row is selected, and the 'currentInteraction' property is highlighted. The bottom section shows the 'Text Cell' configuration panel.

Page Configuration | **Text**

Properties: **Properties** | Reflection | PostOnChange

Text Cell

▼ **Basic properties**

action	
editable	true
id*	InteractionSummary
label	DisplayKey.get("Ext.Summary")
required	
value	currentInteraction.Summary

8. Add the AltUserCell

The screenshot shows the SAP NetWeaver IDE interface for configuring a list view. The top pane displays the 'InteractionsLV.pcf' file with a list view structure. The 'Row Iterator' is set to 'currentInteraction'. The list view has three columns: 'Date', 'Summary', and 'User'. The 'User' column is currently empty. The 'Row' section shows a date field, a 'currentIntera...' field, and an 'AltUser...' field which is highlighted with a blue border.

The bottom pane shows the 'Page Configuration' tab with the 'Text' sub-tab selected. The 'Properties' section is active, showing the properties for the 'AltUserCell'.

Basic properties	
action	
editable	true
id*	AssociatedUser
label	DisplayKey.get("Ext.AssociatedUser")
required	
value	currentInteraction.AssociatedUser

2.6 Lab Solution: Reference the List View



Solution

Exact details on how to complete the lab.

1. Open the **ABContactInteractionsPage** and replace the **DetailView** with a **PanelRef**.

InteractionsLV.pcf x ABContactInteractionsPage.pcf x

Page : ABContactInteractionsPage anABContact

Tab Bar : TabBar

Search Contact Administration Unsaved Work International Example List About Preference

Info Bar : ABContactInfoBar anABContact

[label]: value [label]: value [label]: value [label]: value type.DisplayName\::anABCo

Actions

[ABContactLGMMenuActions]

Menu Links : ABContactLG

Menu Item

Summary

Menu Item

Details

Screen

Panel Ref

List View : InteractionsLV anABContact

Row Iterator currentInteraction

Date Summary User

Row

currentIntera...

AltUserCell : AssociatedU...

Page Configuration Text

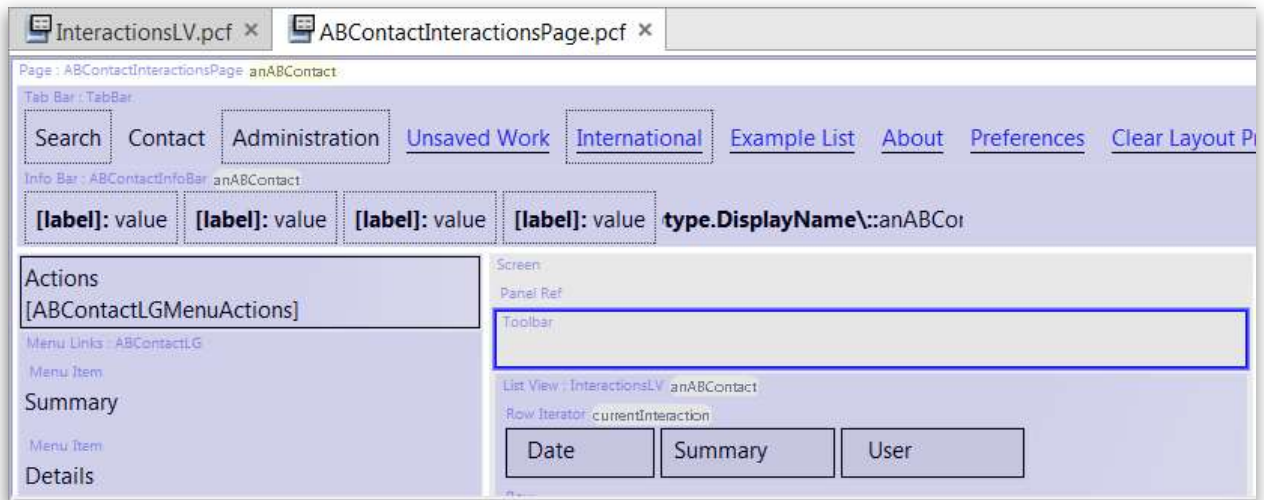
Properties Properties

Panel Ref

Basic properties

def InteractionsLV (anABContact)

2. Add a Toolbar into the PanelRef



2.7 Bonus Lab Solution: List Views and navigation



Solution

Exact details on how to complete the lab.

1. Define the **action** property of the **TextCell** in **ABCompanyEmployeesLV**
2. Set **actionAvailable** to true to allow navigation when not in edit mode

