

Guidewire training materials contain Guidewire proprietary information that is subject to confidentiality and non-disclosure agreements. You agree to use the information in this manual solely for the purpose of training to implement Guidewire software solutions. You also agree not to disclose the information in this manual to third parties or copy this manual without prior written consent from Guidewire. Guidewire training may be given only by Guidewire employees or certified Guidewire partners under the appropriate agreement with Guidewire.

# Lesson objectives

- By the end of this lesson, you should be able to:
  - Identify the parent containers for an Input Set
  - Describe Input Set reuse and shared logic
  - Create a shared logic input set
  - Create a reusable input set
  - Differentiate between an Input Set and Input Set Ref element

This lesson uses the notes section for additional explanation and information.  
To view the notes in PowerPoint, select View → Normal or View → Notes Page.  
When printing notes, select Note Pages and Print hidden slides.

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

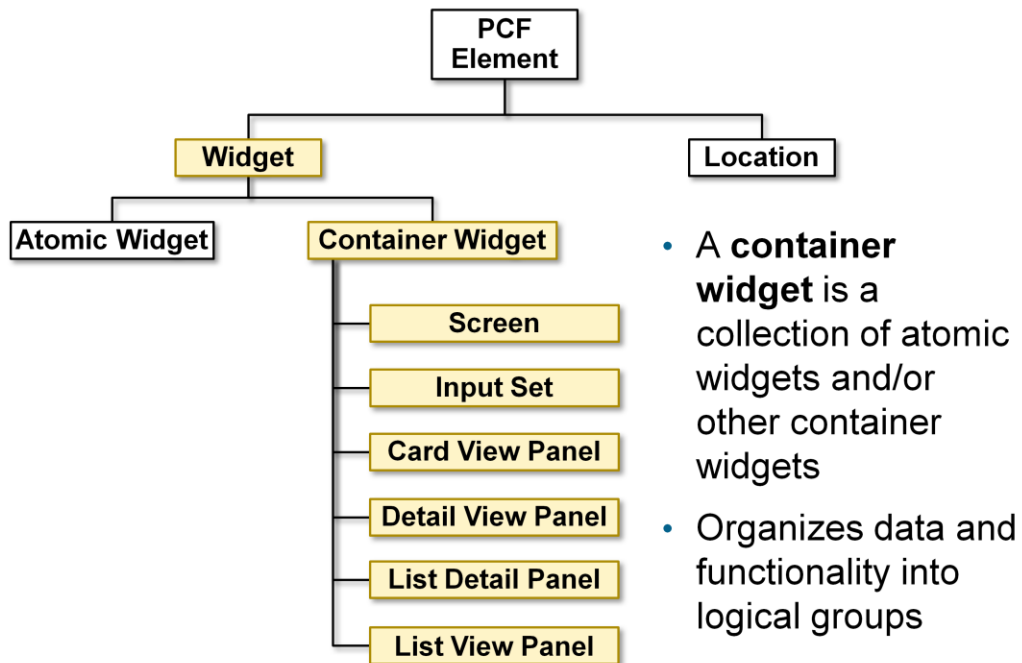
2

G U I D E W I R E

## Lesson outline

- Input set fundamentals
- Shared logic input set
- Reusable input set

# Container widgets



© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

4

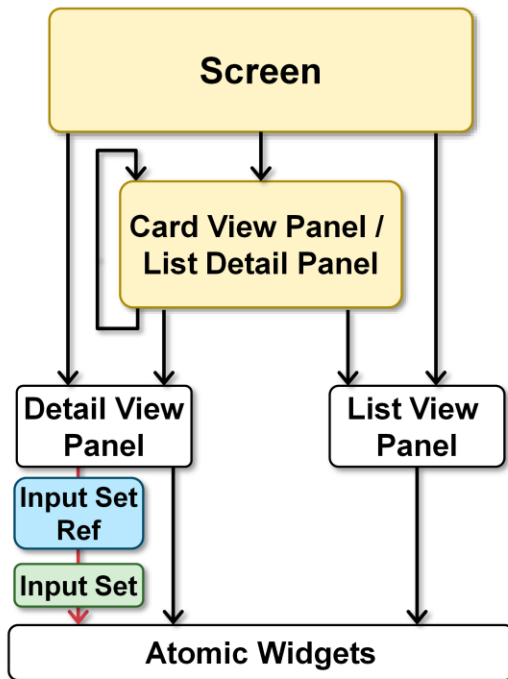
G U I D E W I R E

Container widgets hold other widgets. Each one can be defined either in its own file or as a child container within some other PCF element file.

Both Widget and Location are conceptual representations in this diagram. There are no `<Widget />` or `<Location />` elements. Similarly, both Atomic Widget and Container Widget are conceptual representations. There are no `<Atomic Widget />` or `<Container Widget />` elements.

The PCF object model is container-based. Each screen element is modeled as an object, which may contain other objects. The hierarchical structure simplifies the task of locating and modifying visual elements. Furthermore, each element can be declared as an independent and therefore reusable element.

## Container hierarchy



- A reference widget in a parent container references a PCF File as an embedded child container
- An InputSetRef widget can reference an Input Set

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

5

GUIDEWIRE

An input set can contain child input sets. An input set can be added to an input column defined within a detail view panel. An input set can be referenced within an input column defined within a detail view panel. To reference an input set, you use an InputSetRef widget.

In practice, Studio automatically adds an input column to a detail view panel if you attempt to add an input set where there is not already an input column.

Although not reflected in the diagram, an input set can contain a List View Input that references a list view panel. In practice, there is not a frequent need to include a list view panel in an input set.

# Container widgets: Input Sets

The screenshot shows a web form for a user profile. It has two main sections: 'Basic Information' and 'Primary Address'. The 'Basic Information' section includes fields for Name (Eric Andy), Public ID (ab:98), Created On (12/06/2013), and Assigned User (a dropdown menu showing '<none>'). The 'Primary Address' section includes fields for Country (United States), Address 1 (345 Fir Lane), Address 2, Address 3, City (La Canada), County, State (California), ZIP Code (91352), Address Type (Home), Description, and Valid Until (MM/dd/yyyy). A blue box labeled 'Input Set' is overlaid on the 'Primary Address' section, indicating that this group of fields is an input set.

- An **input set** is a named group of widgets
  - Can contain atomic widgets
  - Can be reused by detail view panels
  - Cannot be referenced by secondary views
  - Cannot have a toolbar directly associated with it
- Reuse single set of inputs across multiple detail view panels
- Apply visibility or editability logic across multiple widgets

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

6

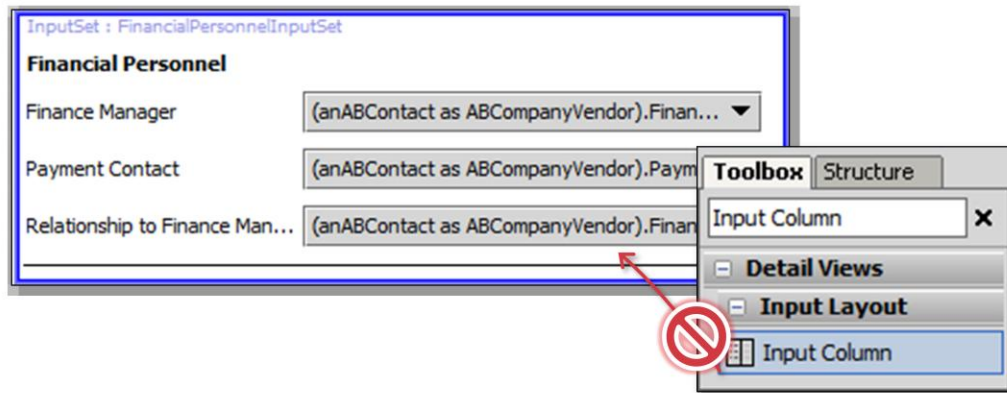
G U I D E W I R E

An input set groups together a set of widgets. There are two types of input sets: InputSet widget and Input Set PCF file.

An InputSet widget groups together input widgets and allow developers to apply shared logic to the grouped widgets. An InputSet widget is an inline widget within an input column of a detail view panel.

An Input Set PCF allows developers to create a group on input widgets within a file. An Input Set PCF can be referenced by one or more PCF files. Input Set PCFs allow for reuse.

# Widgets for input sets



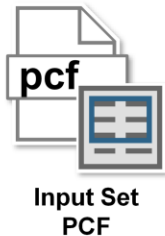
- Input sets can contain
  - Input widgets
  - Input dividers
  - Label widgets
  - List view panels
- Input sets cannot contain
  - Input columns

Unlike detail view panels, input sets cannot contain columns.

## File & widget

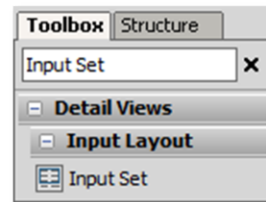
### Input Set PCF file

- `<InputSet />` is a top-level PCF element
- File name ends with `InputSet`
- Can define root object



### InputSet widget

- Widget is defined in an Input Column that is within a Detail View Panel



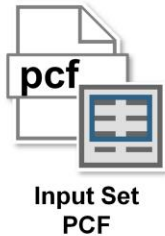
It is possible to define a variable for an InputSet widget. In many cases, however, an InputSet widget inherits the root object associated with its parent Detail View Panel.



# Reusability and inline

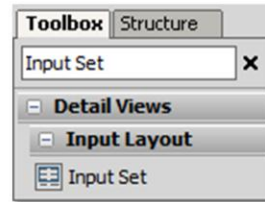
## Reusability

- Reuse single set of inputs across multiple detail view panels
- Input Set is PCF file
- InputSetRef widget references the Input Set PCF file



## Inline is Shared logic

- Apply visibility or editability logic for a group of widgets
- Input Set is often a widget in an input column within a detail view panel



© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

9

G U I D E W I R E

The two use cases for input sets—reusability and shared logic—are not mutually exclusive. It is possible to have an input set that extends a single visible or editable condition across its contents and gets reused under multiple circumstances.

## Lesson outline

- Input set fundamentals
- Shared logic input set
- Reusable input set

## Input set shared logic

- Group the visibility and/or editable property conditions for a set of widgets
- Widgets placed within input set
- Visibility and/or editability set at input set level
- Example:
  - If the root object is of type `ABCompanyVendor`, then `FinancialPersonnelInputSet` is visible

The screenshot shows a web form titled "Auto Repair Shop: Burlingame Saab". It has tabs for "Company Info", "Phone & Addresses", and "Bank Account". The "Company Info" tab is active, showing "Employee Info" with fields for "Can Have Employees?" (radio buttons for Yes/No), "Number of Employees" (3), and a table of employees. Below the table is a section titled "Financial Personnel" which is highlighted with a red box. This section contains three dropdown menus: "Finance Manager" (selected: William David), "Payment Contact" (selected: William David), and "Relationship to Finance Manager" (selected: Self).

Name	Job Title	Email Address
William Dan	Manager	
William David	Finance Accounting	
Luke Lowndes	Technician	

**Financial Personnel**

Finance Manager: William David

Payment Contact: William David

Relationship to Finance Manager: Self

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

11

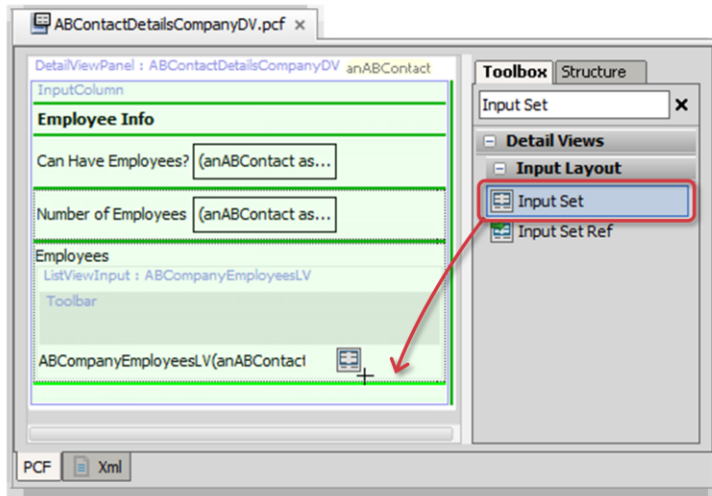
GUIDEWIRE

It is also possible to set an editable or visible property for an `InputSet` widget. The input set property applies to all the contained widgets within the input set. In this manner, you can eliminate the need to configure the individual widget properties for a group of related widgets.

## Steps to create a shared logic input set

1. Add an InputSet widget
2. Specify shared logic
3. Add atomic widgets
4. Deploy PCF

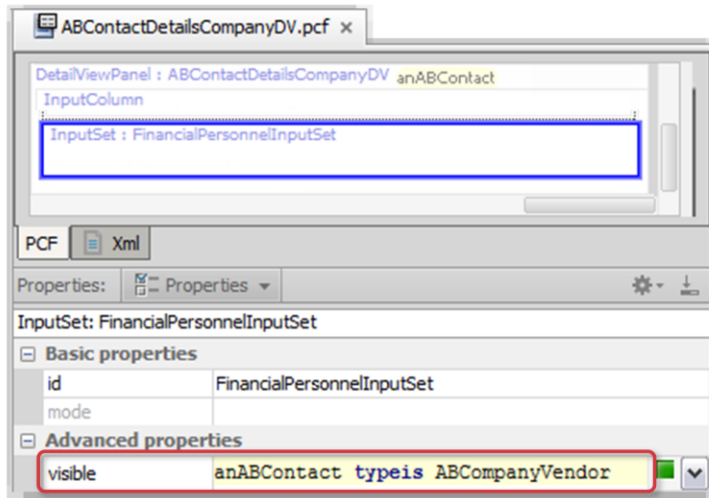
## Step 1: Add an InputSet widget



- Light green line - current place where new widget will go
- Dark green line - places where new widget can go

- Add an InputSet widget
  - In an Input Column within a Detail View Panel
  - Can reference parent container variables

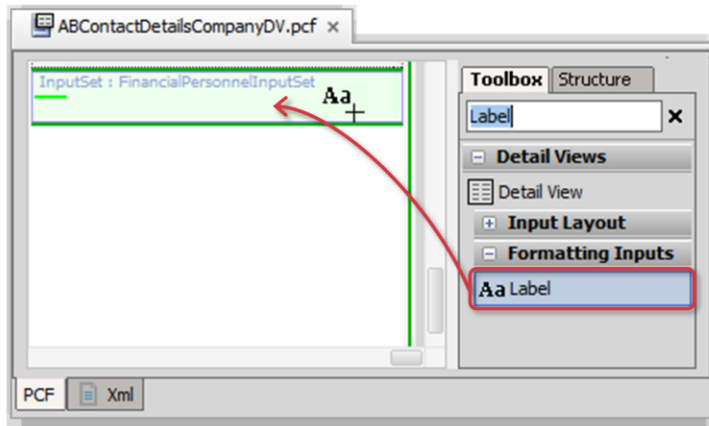
## Step 2: Specify shared logic



- Input set is parent container for its child widgets
- Editable and or Visible properties affect all child widgets of the input set

- Example:
  - Visible is true when object is of the type ABCompanyVendor
  - **typeis** is a Gosu operator to compare object to a type

## Step 3: Add atomic widgets



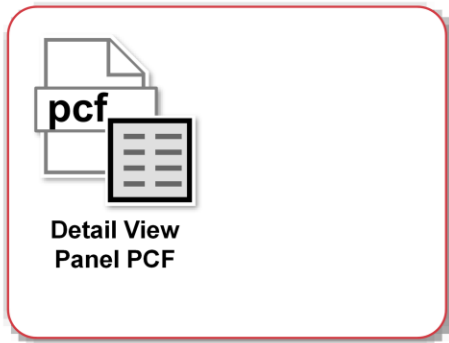
- Select the best input for the data type from the Toolbox
- Specify required and optional properties

Add an atomic widget as you would for a detail view panel.

## Step 4: Deploy PCF

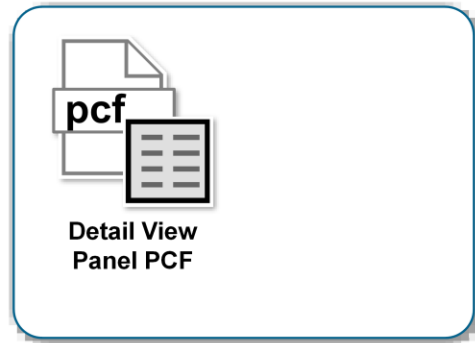
### Restart Server

- PCFs read at server startup



### Reload PCFs

- ALT+SHIFT+L
  - Internal debug tools enabled
- Internal Tools
  - Reload → Reload PCF Files



© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

16

GUIDEWIRE

It is also possible to reload PCF files using the Guidewire API and/or internal server tools. The Reload PCF command can be found on the Reload page in Internal Tools. To access Internal Tools, you must log in as an administrator user, e.g., su/gw. Then, use ALT+SHIFT+T. In the tab bar, select Internal Tools → Reload. On the Reload page, click the Reload PCF Files button. The Reload PCF Files button calls the static method `gw.api.tools.InternalToolsUtil.reloadPCFs()`.





## Lesson outline

- Input set fundamentals
- Shared logic input set
- Reusable input set

# Input set reusability

Country: United States  
Address 1: 345 Fir Lane  
Address 2:   
Address 3:   
City: La Canada  
County:   
State: California  
ZIP Code: 91352

GlobalAddressInputSet.default

Summary

Basic Information

Name: Eric Andy  
Public ID: abc98  
Created On: 12/06/2013  
Assigned User: <none>

Primary Address

Country: United States  
Address 1: 345 Fir Lane  
Address 2:   
Address 3:   
City:   
County: La Canada  
State: California  
ZIP Code: 91352

ABContactSummaryDV

ABContactAddressesLDV

Actions: Update, Cancel, Add, Remove, Delete Secondary Addresses

Summary

Details

Addresses (3)

Notes (0)

Analysis

Interactions

History

Address Detail

Address Type: Home

Description: 345 Fir Lane, La Canada, CA 91352, ...

Country: United States  
Address 1: 345 Fir Lane  
Address 2: 8982 Merrydale Dr, San Francisco, ...  
City:   
County:   
State:   
ZIP Code: 12345

GUIDEWIRE

18

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

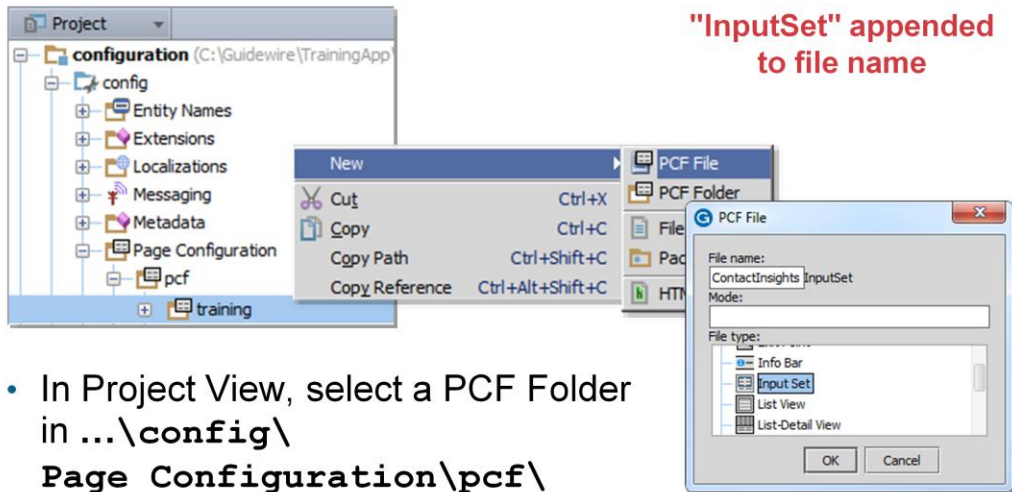
GlobalAddressInputSet is a globalization input set and is modal. Modal widgets are discussed in greater detail later in this course. It is not a requirement for a PCF to be modal in order for it to be reused.

Input sets often define a set of widgets that are reused in multiple detail view panels. Reuse of widgets as an input set reduces and simplifies development, configuration, and administration.

## Steps to create a reusable input set

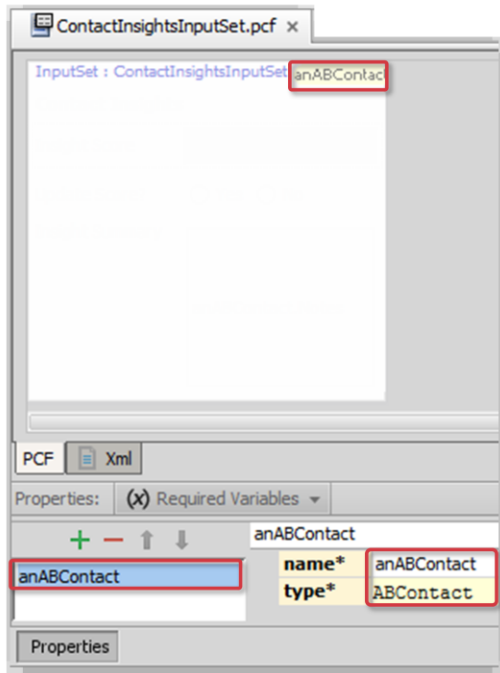
1. Create an input set PCF
2. Specify required variable(s)
3. Add atomic widgets
4. Reference the input set from parent
5. Deploy PCFs

## Step 1: Create an input set PCF



- In Project View, select a PCF Folder in ...\**config**\ **Page Configuration**\pcf\
- Context menu → New → PCF File
- Enter File Name and select Input Set as file type in PCF File dialog

## Step 2: Specify required variable(s)



- Required Variables tab
  - Defines data object variable name and type
  - Example:  
anABContact is of type ABContact
- Object data can be
  - Data backed (database)
  - Virtual property
- Container data comes from defined variable object(s)
  - Typically, at least one variable
  - Not required



© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

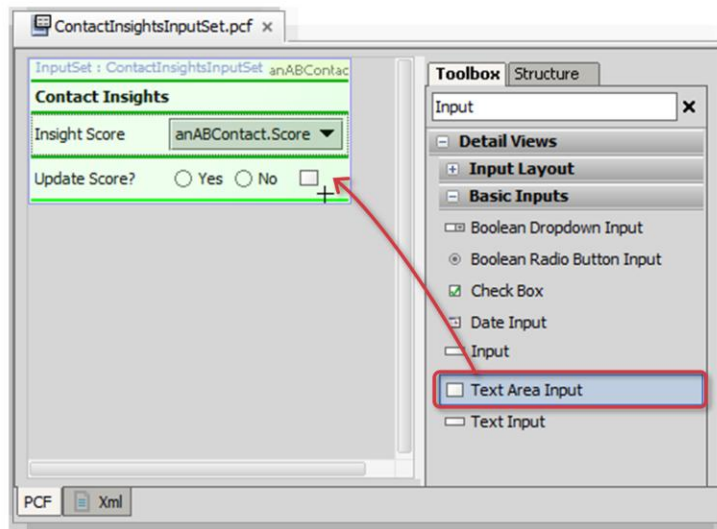
21

G U I D E W I R E

Most container widgets have at least one required object that contains data fields. One way to think of this is that there is at least one root object for a given container.

It is possible to have more than one defined object as it is also possible to not have a required object at all.

## Step 3: Add atomic widgets



- Select the best input for the field data type from the Toolbox
- Specify required and optional properties

- If required, bind the widget data
  - Root object field, subtype object field, related object field

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

22

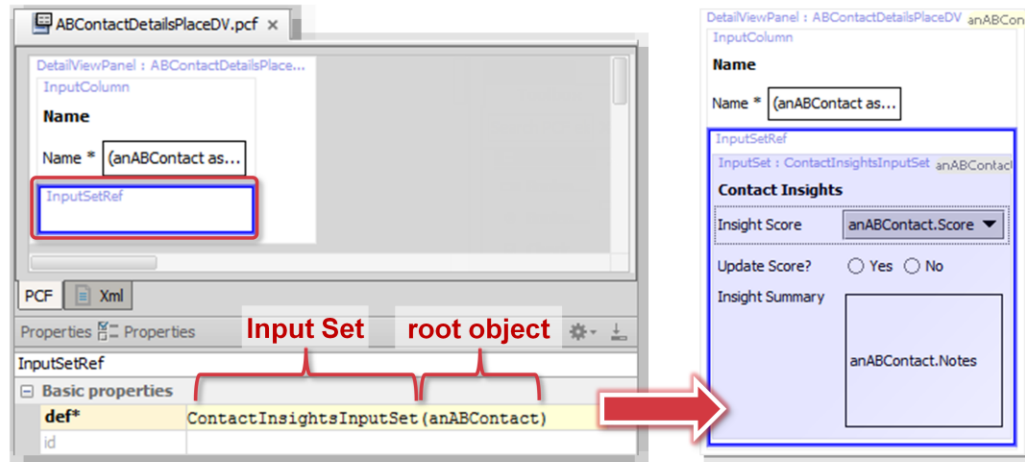
G U I D E W I R E

Atomic widgets are added to an input set in the same way that they are added to a detail view panel. To add a widget, click its name in the Toolbox and hold the mouse cursor down. As you begin to drag the widget, Studio changes the mouse cursor so that it includes the icon for that widget. Studio places a green line on the canvas at every location on the canvas that it is possible to place the widget. Studio highlights the green line that is nearest on the canvas to the cursor. Studio also overlays in green the element containing the highlighted green line.

All PCF elements have definable properties in the Properties window. To view properties of a PCF file, click its title link in the upper-left corner. To view properties of any element, click that element. The Properties window contains multiple property tabs. Click a tab to edit the associated properties. Some properties are not editable. Other properties are required. Required properties have an asterisk and the property name appears against a yellow background. If you select a property, variable, or entry point, an "X" icon appears on the right-hand side of the cell for that property, variable, or entry point. You can click the "X" to restore the selected property, variable, or entry point to its default value. The Properties window validates each property expression and/or value.

Most atomic widgets require IDs. However, some widgets, such as labels and dividers, do not require IDs. The value property defines the data field. You specify a field using dot notation. You can reference a direct object field or related object field. If the field is a "data" field, you can set the editable property of the widget to "true". Only data fields are editable.

## Step 4: Reference the input set from parent



- Add an InputSetRef widget to the parent container
- Define the def property
  - Specify the input set
  - Pass the required object type as an argument

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

23

GUIDEWIRE

A PanelRef widget references a detail view panel. In a similar manner, an InputSetRef widget references an input set. An input set can be added or referenced only inside a detail view panel input column. The PCF Editor in Guidewire Studio automatically adds an input column to a detail view panel if you attempt to add an input set ref where there is not already an input column.

To reference an input set from a parent container:

- Add an InputSetRef widget at the appropriate place in the parent container.
- In the input set ref's def property, specify the input set name. After the name, inside parentheses, specify the required object(s) to pass to the input set.

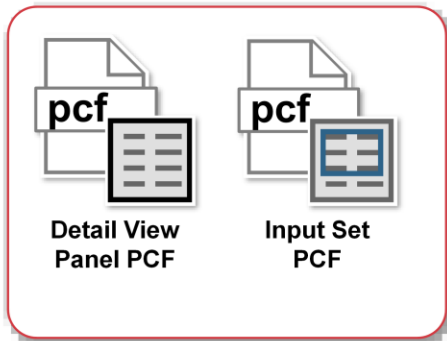
In the slide example, ABContactDetailsPlaceDV defines an root object named anABContact. ABContactDetailsPlaceDV contains a newly added Input Set Ref. The Input Set Ref requires a value for the def property. The def property references the Input Set named ContactInsightsInputSet. The def property passes the anABContact root object as argument to ContactInsightsInputSet.

Similar to the Input Set itself, an Input Set Ref has editable and visible widget properties. Setting these properties affect the referenced Input Set.

## Step 5: Deploy PCFs

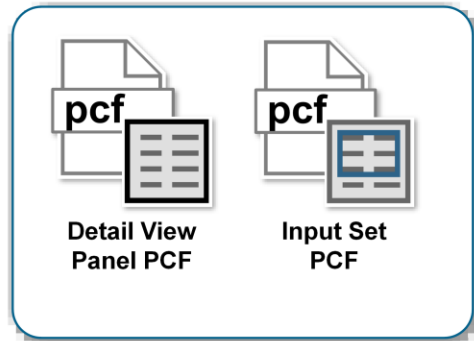
### Restart Server

- PCFs read at server startup



### Reload PCFs

- ALT+SHIFT+L
  - Internal debug tools enabled
- Internal Tools
  - Reload → Reload PCF Files



© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

24

G U I D E W I R E

It is also possible to reload PCF files using the Guidewire API and/or internal server tools. The Reload PCF command can be found on the Reload page in Internal Tools. To access Internal Tools, you must log in as an administrator user, e.g., su/gw. Then, use ALT+SHIFT+T. In the tab bar, select Internal Tools → Reload. On the Reload page, click the Reload PCF Files button. The Reload PCF Files button calls the static method `gw.api.tools.InternalToolsUtil.reloadPCFs()`.



## Lesson objectives review

- You should now be able to:
  - Identify the parent containers for an Input Set
  - Describe Input Set reuse and shared logic
  - Create a shared logic input set
  - Create a reusable input set
  - Differentiate between an Input Set and Input Set Ref element

## Review questions

1. What are the two primary use cases for input sets?
2. Which of the following can be included in input sets:
  - a) Embedded list views
  - b) Inputs
  - c) Input columns
  - d) Input dividers
  - e) Labels
3. What is the difference between the "input set" widget and the "input set ref" widget?

© Guidewire Software, Inc. 2001-2014. All rights reserved. Do not distribute without permission.

26

GUIDEWIRE

### Answers

1) Reuse of widgets across detail view panels and/or logic shared for a group of widgets such as editability and/or visibility.

2a) Yes

2b) Yes

2c) No

2d) Yes

2e) Yes

3) The InputSet widget groups widgets together and is placed within an input column within a detail view panel. You use an InputSetRef widget to reference an Input Set PCF file. An Input Set PCF can be referenced by one or more InputSetRef widgets. Input Set PCFs are reusable input sets.

# Notices

**Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.**

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

This file and the contents herein are the property of Guidewire Software, Inc. Use of this course material is restricted to students officially registered in this specific Guidewire-instructed course, or for other use expressly authorized by Guidewire. Replication or distribution of this course material in electronic, paper, or other format is prohibited without express permission from Guidewire.

Guidewire products are protected by one or more United States patents.