

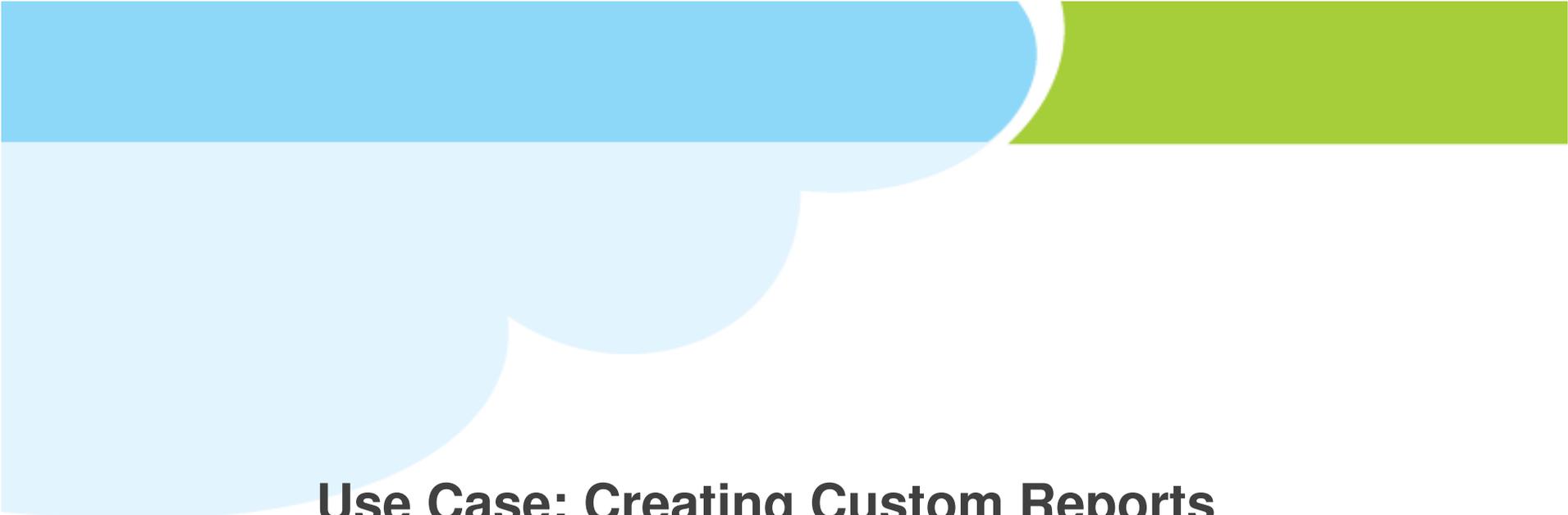


Leading the **Integration** Revolution

Your business problems have changed.
Why hasn't your integration solution?



ADEPTIA



Use Case: Creating Custom Reports

High level scenario

In this example we will create a custom Wire Transfer report based on an Account ID which is passed as a parameter through a web form. This Use Case consists of following items:

1. Design a report template in iReport designer
2. Integrate the report template into Adeptia
3. Execute the report generation dynamically by passing a parameter in an orchestration

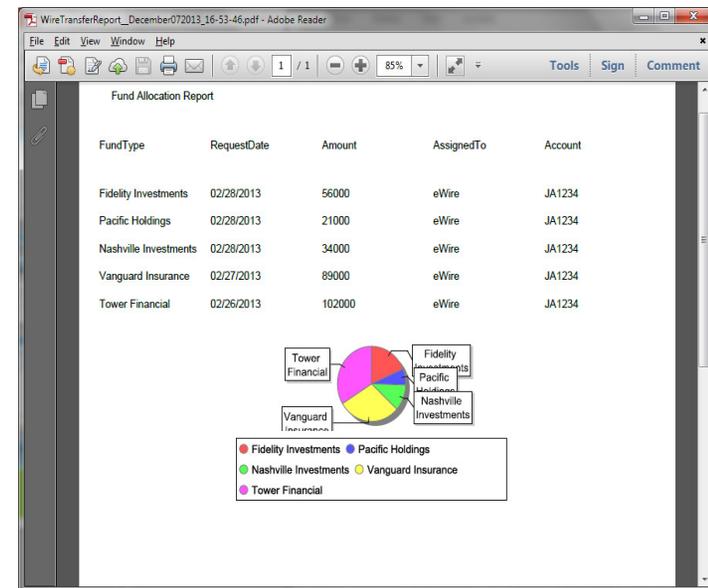
Refer to the Use Case video and the deployable solution file for more information.

Requirement: iReport + Adeptia Suite install with license key that includes Custom Report.

Wire Transfer Report

Wire Transfer data

FundType	RequestDate	Amount	AssignedTo	Account
Fidelity Investments	02/28/2013	56000	eWire	JA1234
Pacific Holdings	02/28/2013	21000	eWire	JA1234
Nashville Investments	02/28/2013	34000	eWire	JA1234
Vanguard Insurance	02/27/2013	89000	eWire	JA1234
Tower Financial	02/26/2013	102000	eWire	JA1234
Pimco Index	07/12/2013	30000	eNet	RS5678
Chase	07/19/2013	9000	eNet	RS5678





Setup of iReport and a report template

Setup Step 1: Download and install iReport

<http://sourceforge.net/projects/ireport/files/iReport/iReport-5.1.0/>

Note: link may change so a better option is to search 'download iReport'.
If you have iReport already installed skip to Step 2.

Home / Browse / Development / Build Tools / iReport-Designer for JasperReports / Files



iReport-Designer for JasperReports

Brought to you by: [gt78](#), [jfsupport](#), [mgeise](#)

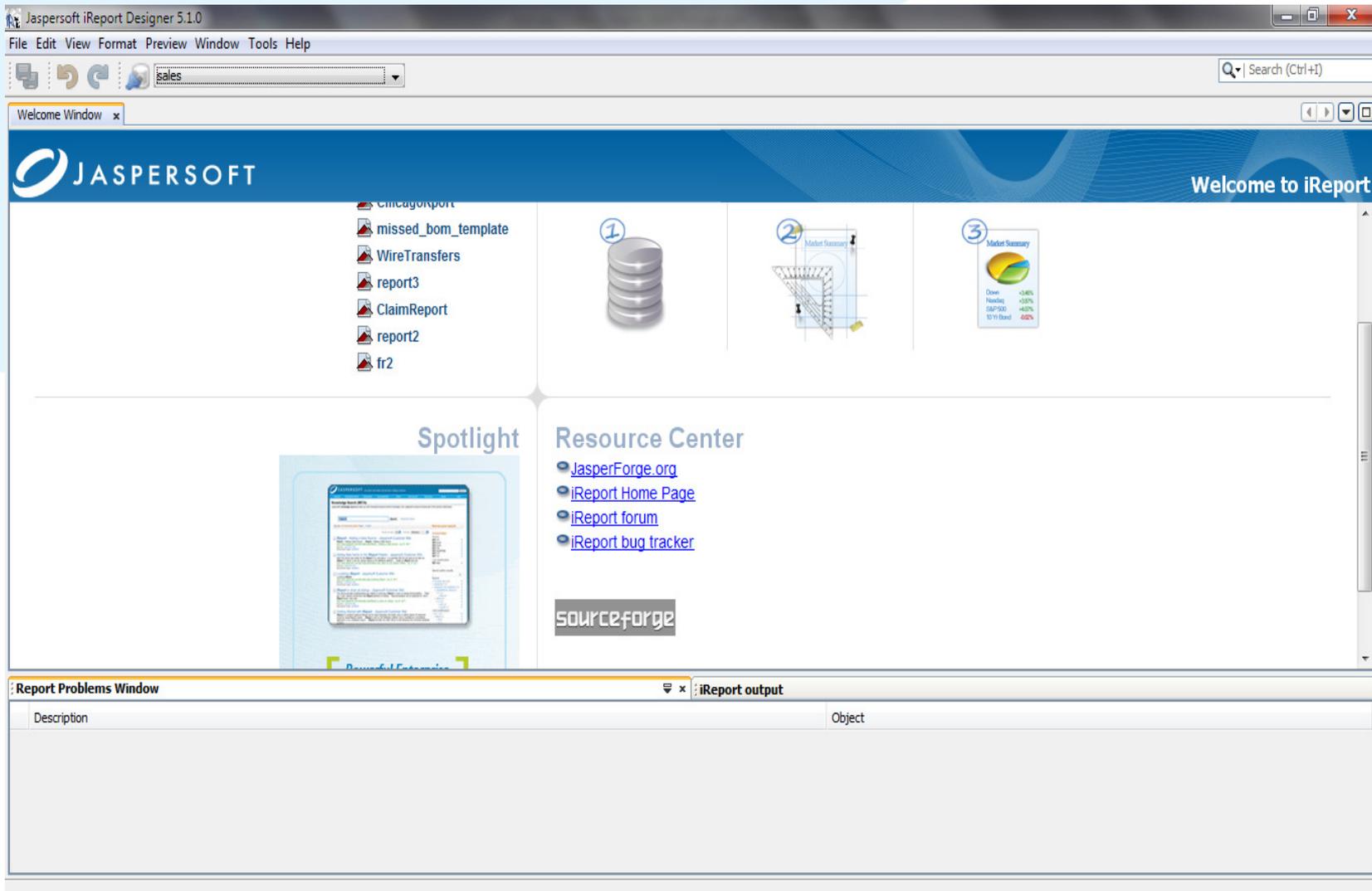
Summary | **Files** | Reviews | Support | Wiki | Piwik | News | Discussion | Donate

Looking for the latest version? [Download iReport-5.5.0-windows-installer.exe \(100.8 MB\)](#)

Home / iReport / iReport-5.1.0

Name ↕	Modified ↕	Size ↕	Downloads / Week ↕
↑ Parent folder			
✓ iReport-5.1.0-windows-installer.exe	2013-04-29	98.0 MB	412  
iReport-5.1.0.zip	2013-04-29	98.6 MB	136  
iReport-5.1.0.tar.gz	2013-04-29	97.5 MB	29  
iReport-5.1.0.dmg	2013-04-29	101.2 MB	18  

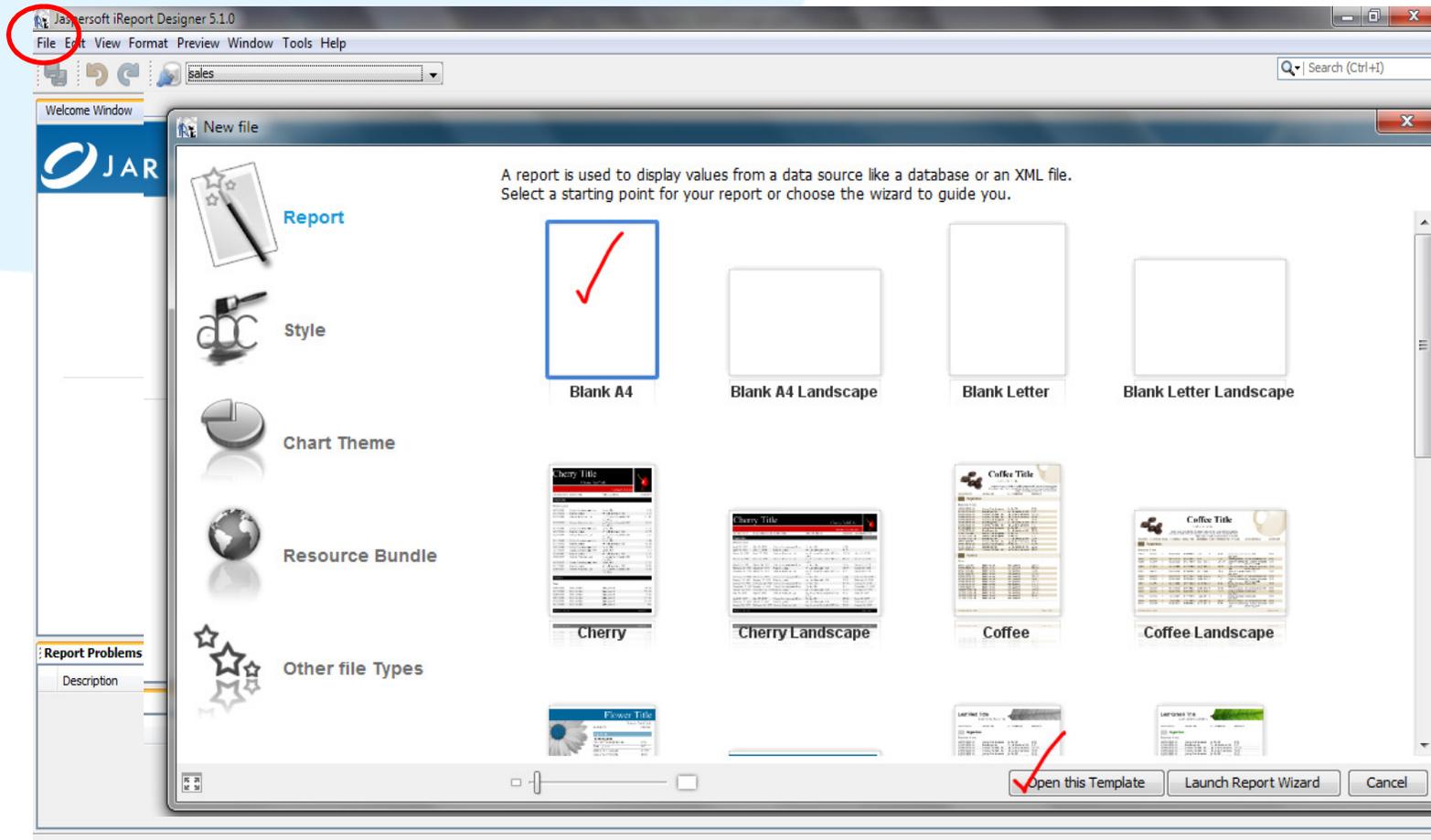
Setup Step 2: Open iReport Designer



Setup Step 3: Let's first setup a blank template

Click on File (top left) and then New.

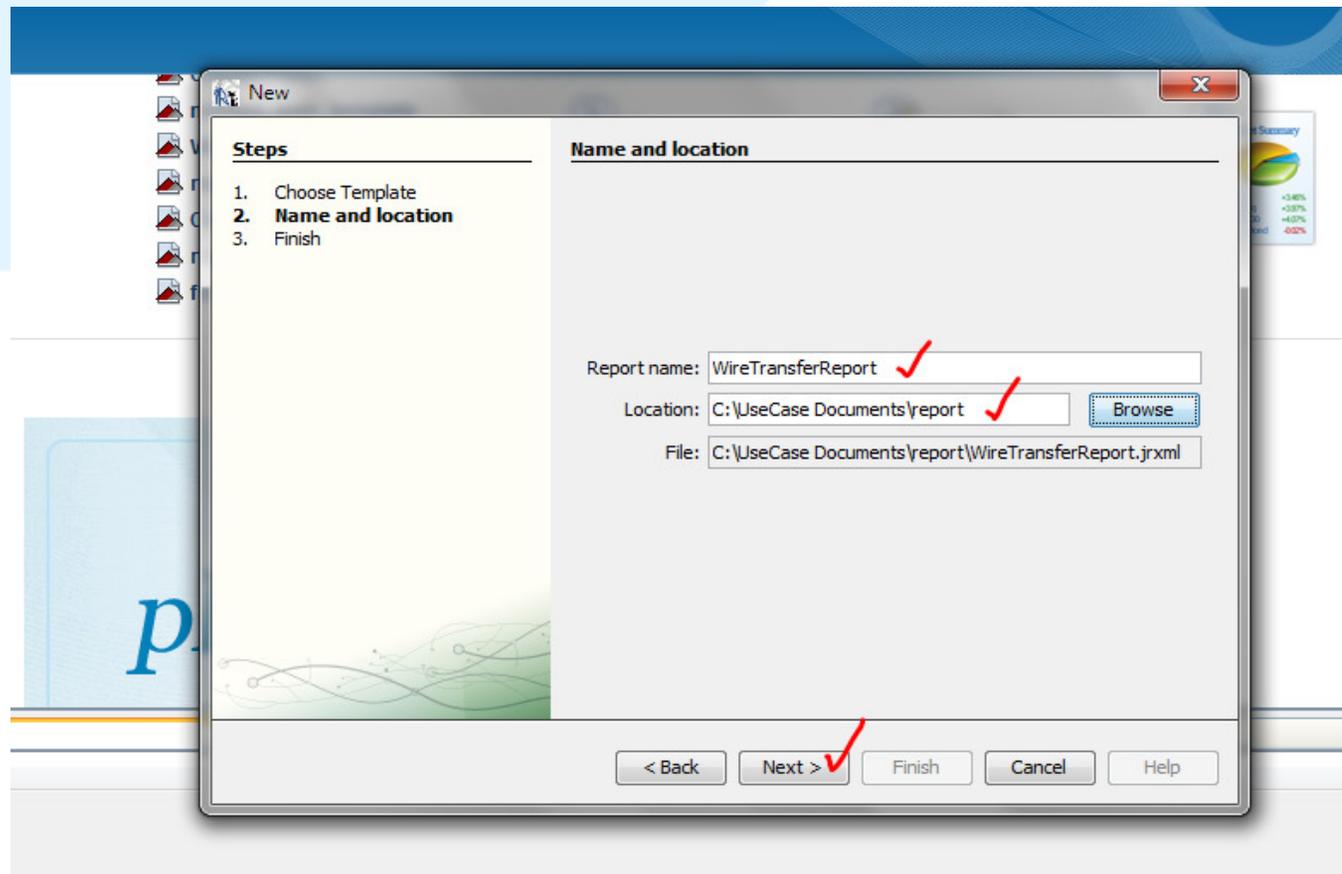
Select the 'Blank A4' and click on 'Open this Template' button located at the bottom.



Setup Step 4: Give a name and path for your report template

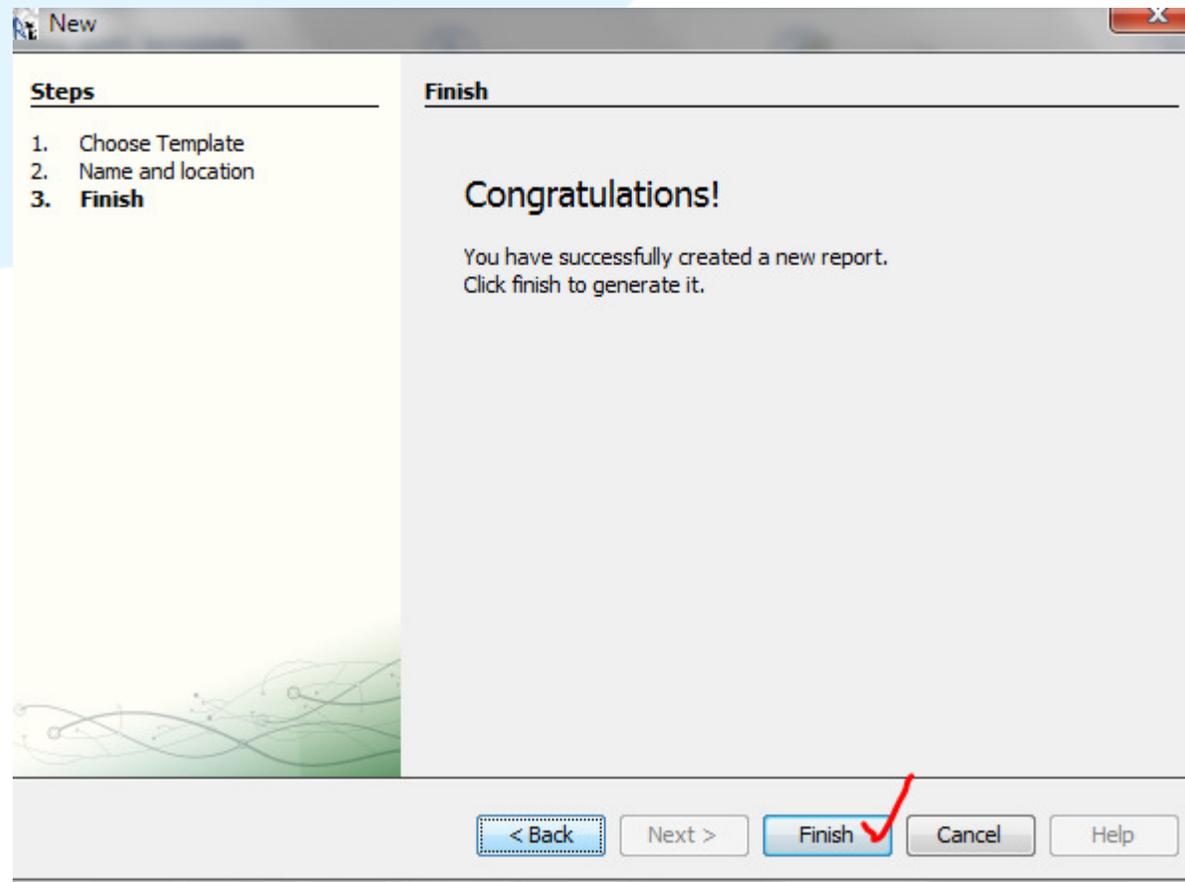
The name and path can be configured in this screen as shown here.

For example this report has a name 'WireTransferReport' and its template is stored at the path selected in the 'Location' field. Click Next.



Setup Step 5: Click Finish and the template is ready for design

Click Finish.

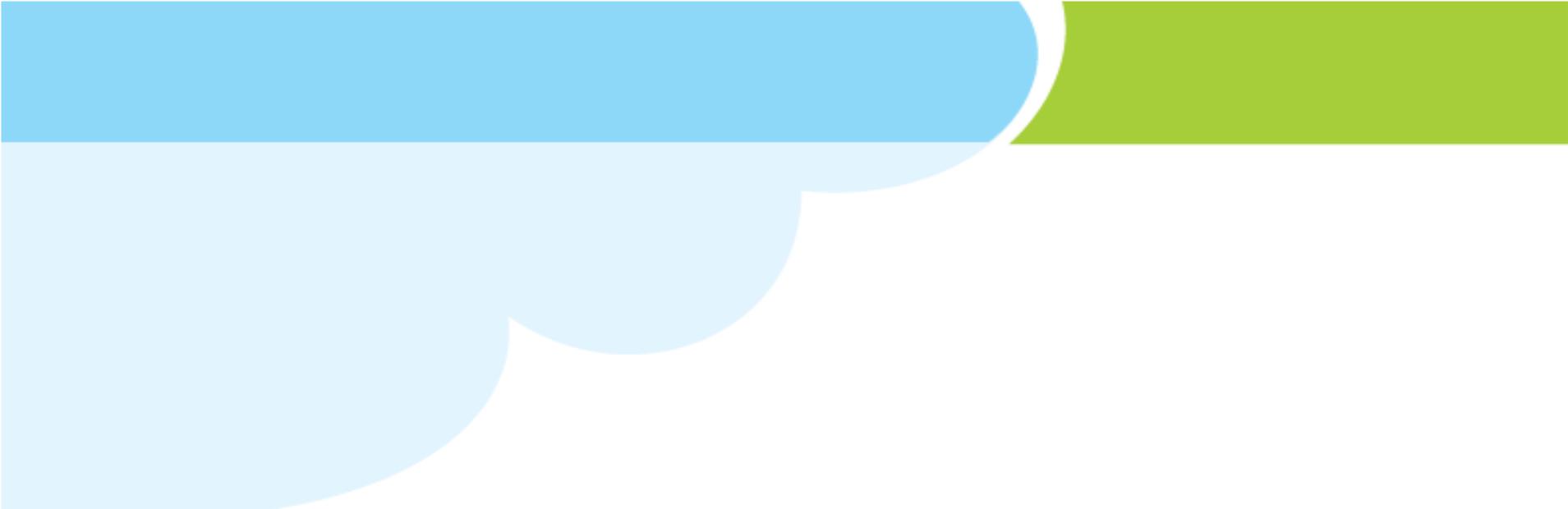


Setup Step 6: Change General properties of the template

Change the 'Language' property of the Report by clicking on the area **above Title** and then Going to the Properties panel and change the Language from 'Groovy' to 'Java'.

Note: For Adeptia version prior to 6.1 you would need to change the compatibility mode of this Report by going to Tools > Options > Under iReport click on Compatibility tab and select 3.5.1 version from the list.

The screenshot displays the Adeptia Designer interface for editing a report template. The main workspace shows a grid-based layout with sections labeled: Title, Page Header, Column Header, Detail 1, Column Footer, Page Footer, and Summary. A red checkmark is visible above the 'Title' section. The right-hand side features a 'Palette' with various design elements and a 'WireTransferReport - Properties' panel. In the Properties panel, the 'Language' property is set to 'Java', indicated by a red arrow pointing to the dropdown menu. Other properties like 'Column Direction' (Left to Right) and 'When No Data' (No Pages) are also visible. The interface includes a top toolbar with 'Designer', 'XML', and 'Preview' modes, and a ruler at the top.



Custom Report template Design

Design Step 1: Understand the layout of the template

The screenshot displays the iReport Designer interface for a report named 'WireTransferReport.jrxml'. The central workspace shows a report layout with sections: Title, Column Header, Detail 1, Column Footer, and Page Footer. A 'Report Inspector' window on the left lists these sections. A 'Palette' window on the right shows various report elements like Break, Crosstab, Frame, Image, etc. A 'No Properties' window is also visible. A ruler at the top indicates the design area. Three callout boxes provide additional information:

- Drag the divider bars to adjust space between sections**: A callout box with a double-headed vertical arrow pointing to the horizontal lines separating the sections.
- These sections would contain non-iterative data. Refer to later sections as we design this template further.**: A callout box with an arrow pointing to the 'Title', 'Column Header', and 'Page Footer' sections.
- Detail section would contain the iteration of data rows selected from source data.**: A callout box with an arrow pointing to the 'Detail 1' section.

At the bottom, a text box states: "If you do not see the Report Inspector and Palette windows, go to 'Window' tab and select these options from the menu. Refer to iReport Help for more information."

Design Step 2: Configure source database connection

Click on Report Datasource button and create your **default database connection**. This connection will be used to extract schema and data based on a Sql statement.

Data source in this example is a MySQL database. There are jdbc drivers for all databases in the Driver drop down list.

Name	Datasource type	Default
Empty datasource	Empty data source	<input type="checkbox"/>
Sample Database (HS...	Sample Database Co...	<input type="checkbox"/>
Employee Benefits	Microsoft Excel (xls) ...	<input type="checkbox"/>
Financials	Database JDBC conn...	<input type="checkbox"/>
Prism_Data_Connect...	Database JDBC conn...	<input type="checkbox"/>
mortgage	Database JDBC conn...	<input type="checkbox"/>
sales	Database JDBC conn...	<input type="checkbox"/>
sprintdb	Database JDBC conn...	<input type="checkbox"/>
WireTransfer	Database JDBC conn...	<input checked="" type="checkbox"/>

Database JDBC connection configuration:

- Name: WireTransfer
- JDBC Driver: MySQL (com.mysql.jdbc.Driver)
- JDBC URL: jdbc:mysql://localhost:3306/wiretransfer
- Username: root
- Password: [masked]
- Save password:

Connection test successful!

Design Step 3: Configure Report query

2. Write a query and drag the parameter from the Parameter list into this query as shown here.

```
select * from transfer where Account = ${Account}
```

1. Create Parameter

Field name	Field type	Description
FundType	java.lang.String	
RequestDate	java.lang.String	
Amount	java.lang.Integer	
AssignedTo	java.lang.String	
Account	java.lang.String	

3. Query would show the schema of the table below with all the field names that can be now used in the template. Click OK.

Design Step 4: Drag the field names into the Detail section

Press SHIFT key and select all the fields under the 'Fields' list and drag them **into the Detail area** of the report canvas. It will automatically show all the fields in the Detail area along with the field names in the Column header area as shown below. You can also drag each field individually **into the Detail area**.

The screenshot displays the Report Designer interface. On the left, the 'Report Inspector' shows a tree view with 'Fields' expanded, listing 'FundType', 'RequestDate', 'Amount', 'AssignedTo', and 'Account'. The main canvas shows a report layout with sections: Title, Page Header, Column Header, and Detail 1. The 'Detail 1' section contains a table with columns: FundType, RequestDate, Amount, and AssignedTo. Below the column headers, the data rows are represented by expressions: `$F{FundType}`, `$F{RequestDate}`, `$F{Amount}`, and `$F{AssignedTo}`. A callout box with the text 'Select and drag into the Detail section.' points to the 'Detail 1' area. On the right, a 'Palette' window is open, showing various report elements. The 'Text Field' element is highlighted with a red checkmark.

You can also drag 'Text field' from the Palette into the canvas and configure the field name inside the brackets `$F{field}` by right clicking on the field and using edit expression option.

Design Step 5: Let's add a Title

Drag the 'Static Text' from the Palette into the Title section and change the font type or size. Make sure the entire text is visible as you make text bigger by expanding the edges of the text box.

The screenshot shows a report designer interface for a report titled 'WireTransferReport.jrxml'. The main design area displays a report layout with sections: Title, Page Header, Column Header, Detail 1, Column Footer, and Page Footer. The 'Title' section contains the text 'Fund Allocation Report', which is highlighted with a yellow border and red checkmarks. The 'Page Header' section contains a table with columns: FundType, RequestDate, Amount, AssignedTo, and Account. The 'Detail 1' section is currently empty. The 'Palette' on the right side of the interface shows various report elements, with 'Static Text' selected and marked with a red checkmark. The 'Properties' window for the selected 'Fund Allocation Report' shows the following dimensions:

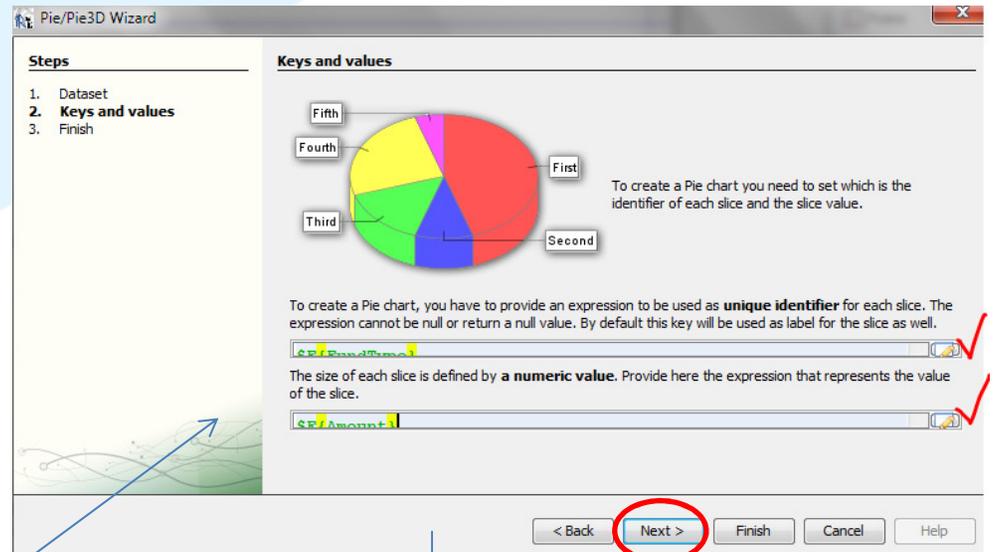
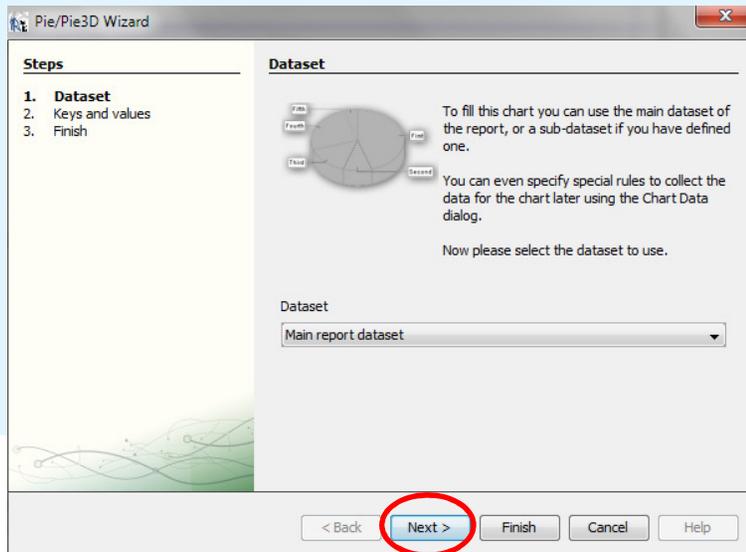
Property	Value
Left	12
Top	10
Width	145
Height	20
Forecolor	Color

Design Step 6: Let's add a Chart

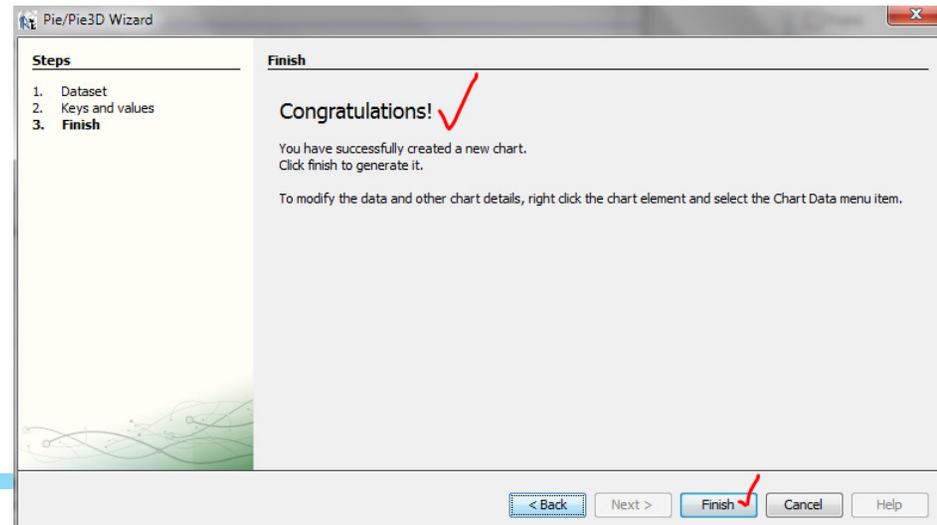
Expand the Summary section and click on Chart in the Palette. In this example we have chosen Pie chart and click OK. Drag the divider bars of the other sections closer to each other and make space for the Summary section where the Pie chart will be inserted.

The screenshot shows a report design tool interface. The main window displays a report titled "Fund Allocation Report" with a table containing columns for "FundType" and "Request". A red double-headed vertical arrow indicates the space being prepared for a chart. A "Chart" dialog box is open, showing various chart types including pie charts, bar charts, and line charts. The "Pie" chart type is selected in the "Chart info" section at the bottom of the dialog. The "Palette" on the right side of the interface shows the "Chart" element selected with a red checkmark. The "Summary - Properties" window is also visible, showing properties for the selected chart.

Design Step 5: Let's add a Chart (contd.)



Select a unique key (string) in the first field.
Select a numeric field in the second field.



Design Step 5: Let's add a Chart (contd.)

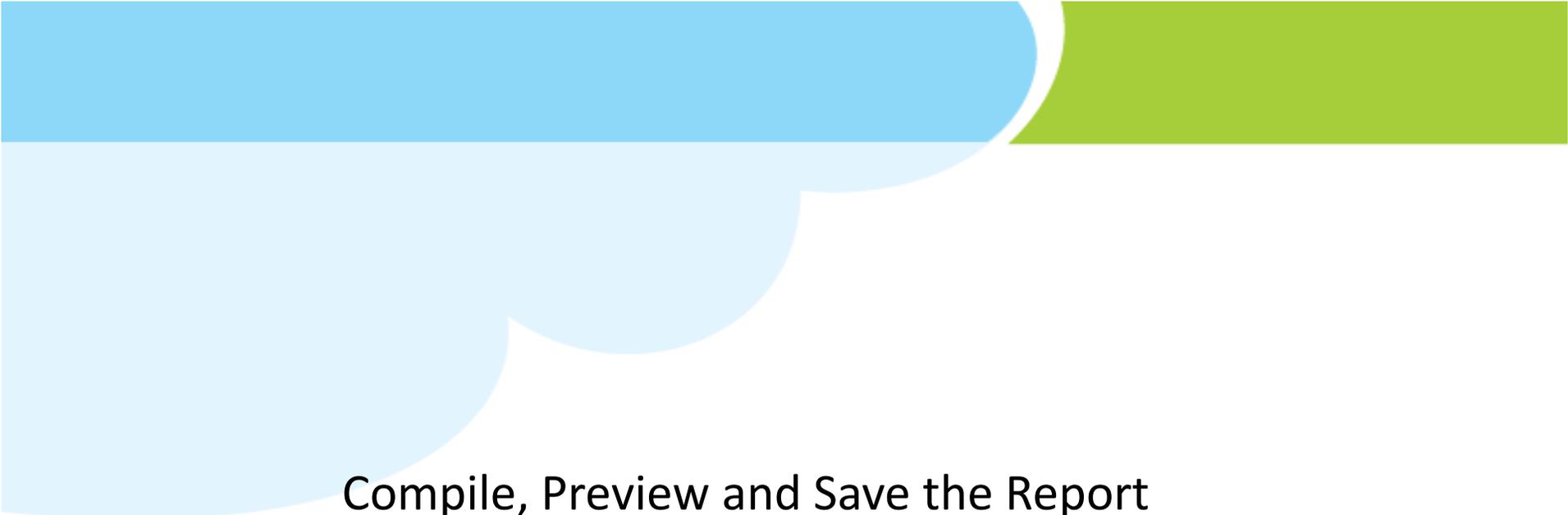
Pie chart is now added to the Summary section. Widen the Summary area by moving the dividers and make the pie chart fit in the section as shown below. If the edges of any component are highlighted with red borders it means that the component is not fitting into the section and this will create errors when compiling the report.

The screenshot shows a report designer interface for a report titled "WireTransferReport.jrxml". The report layout includes a title "Fund Allocation Report", a page header, a table with five columns, and a pie chart in the summary section. The table columns are FundType, RequestDate, Amount, AssignedTo, and Account. The pie chart is divided into five segments labeled First, Second, Third, Fourth, and Fifth. A legend below the chart identifies the segments by color: First (red), Second (blue), Third (green), Fourth (yellow), and Fifth (pink).

FundType	RequestDate	Amount	AssignedTo	Account
<code>#{FundType}</code>	<code>#{RequestDate}</code>	<code>#{Amount}</code>	<code>#{AssignedTo}</code>	<code>#{Account}</code>

Legend:

- First (Red)
- Second (Blue)
- Third (Green)
- Fourth (Yellow)
- Fifth (Pink)



Compile, Preview and Save the Report

Compile the Report

Click on the Compile Report highlighted below and see the result at the bottom.

The screenshot displays a report designer window titled 'WireTransferReport.jrxml'. The main area shows a report layout with a title 'Fund Allocation Report', a table with columns 'FundType', 'RequestDate', 'Amount', 'AssignedTo', and 'Account', and a pie chart with five segments labeled 'First' through 'Fifth'. The 'Compile Report' icon in the toolbar is circled in red. Below the report, the 'Report Problems Window' shows the following output:

```
Report Problems Window
iReport console Finished [WireTransferReport.jrxml]
Compiling to file... C:\UseCase Documents\report\WireTransferReport.jasper
Compilation running time: 356!
Filling report...
Locale: English (United States)
Time zone: Default
```

Preview the Report

Click on the Preview button and verify the results.

The screenshot shows a software interface with a report preview window. The report contains a table of wire transfer data and a pie chart. The 'Preview' button is highlighted with a red checkmark.

Company	Date	Amount	Method	Reference
Fidelity Investments	02/28/2013	56000	eWire	JA1234
Pacific Holdings	02/28/2013	21000	eWire	JA1234
Nashville Investments	02/28/2013	34000	eWire	JA1234
Vanguard Insurance	02/27/2013	89000	eWire	JA1234
Tower Financial	02/26/2013	102000	eWire	JA1234

The pie chart displays the distribution of wire transfer amounts by company:

- Fidelity Investments (Red)
- Pacific Holdings (Blue)
- Nashville Investments (Green)
- Vanguard Insurance (Yellow)
- Tower Financial (Pink)

Legend:

- Fidelity Investments
- Pacific Holdings
- Nashville Investments
- Vanguard Insurance

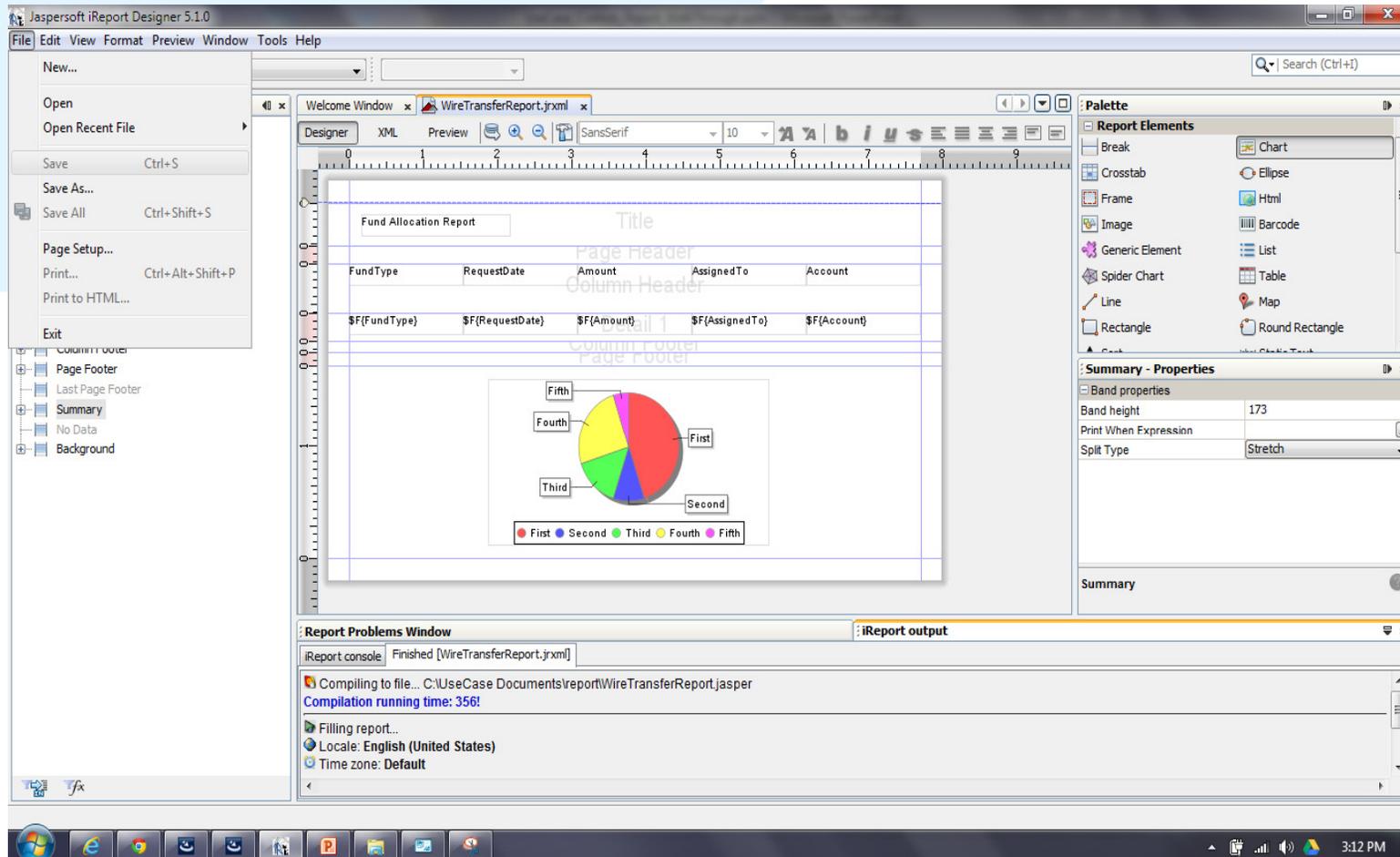
Problems Window: Finished [WireTransferReport.jrxml]

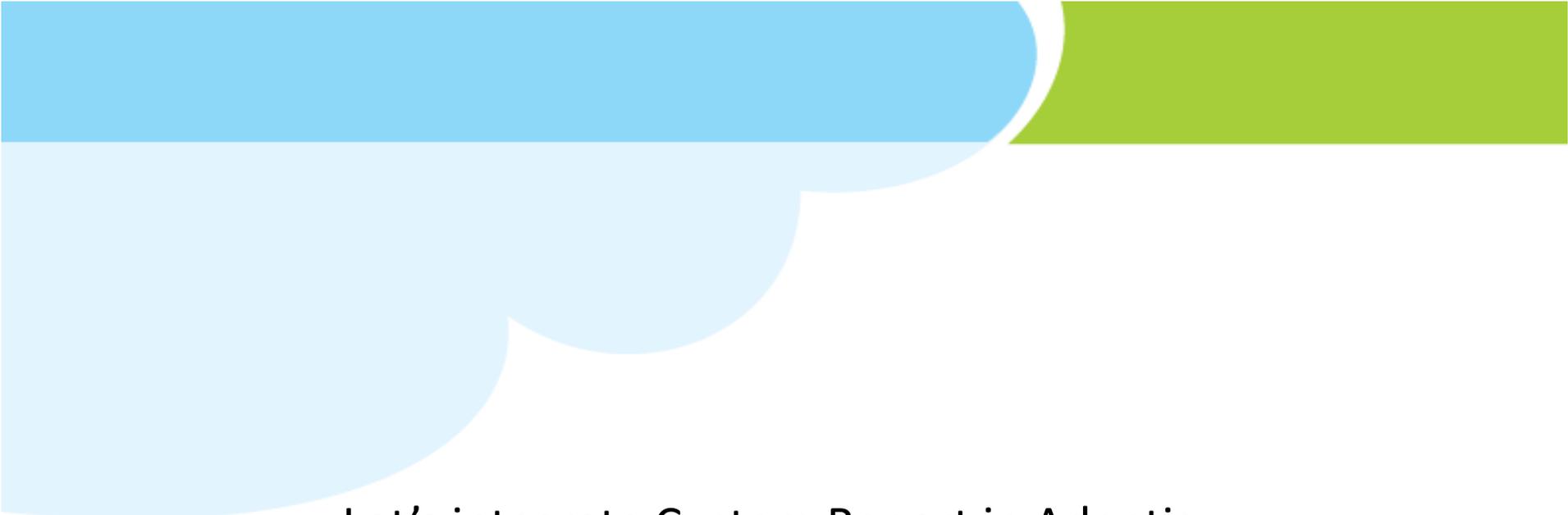
Compilation to file... C:\UseCase Documents\report\WireTransferReport.jasper
Compilation running time: 356!

Generating report...
Locale: English (United States)

Save the Report

After previewing the report, click on the Designer button to go back to the designer window. Click on File > Save. The JRXML report type is saved in the location provided in Setup Step 4.





Let's integrate Custom Report in Adeptia
(Login to Adeptia)

Configure: Custom Report activity

In Adeptia Suite, go to Develop > Reports and Dashboards > Custom Report and click on Create New. As a prerequisite create a DB Info with same database connection as defined in the custom report's data source. Select this DB Info from the list. Upload the JRXML file saved in the previous step.

New Custom Report

Standard Properties

Name* WireTransferReport

Description* wire transfer report

Database Info wiretransfer

Upload JRXML File(s)* WireTransferReport.jrxml

Report Type* PDF

Note: Saving of activity will take time incase multiple JRXML files are uploaded.

Advanced Properties

* Mandatory fields.

Upload Jrxml Files - Windows Internet Explorer

http://localhost:8080/adeptia/uploadjrxmlfile.jsp?initialRequest=true&id=n

- 1 Browse/Choose File(s)**
Click on "Browse/Choose" button to choose the file.
Browse/Choose File
- 2 Upload File**
Click on the "Upload File" button. Please wait till your file appears t
Name.

File Name
[Select All](#) | [Unload File\(s\)](#)
 WireTransferReport.jrxml
- 3 Finish**

Configure: Custom Report activity (parameter value)

Click on the User Defined Parameter and enter the context variable in the Parameter Value.

If you provide a static value then you can generate the report manually by going to Monitor > Reports > Custom Reports and clicking on the Execute option under the Action. Here \$\$Account\$\$ means that the Parameter value will be passed to this activity at run-time in the process flow. Click on Save.

New Custom Report

Standard Properties

Name*	WireTransferReport
Description*	wire transfer report
Database Info	wiretransfer
Upload JRXML File(s)*	WireTransferReport.jrxml
	User Defined Parameter
Report Type*	PDF

Note: Saving of activity will take time incase multiple JRXML files are upload

Advanced Properties

* Mandatory fields.

Jasper Report Parameters

Parameter Name	Parameter Value
Account	\$\$Account\$\$

OK Close

Configure: File Target to store custom reports

Create a Target for your report destination. This can be a File, Email, FTP etc. Example below is a File Target activity.

Name*	ReportTarget		
Description*	place report in folder		
File Path*	c:\		
File Name*	WireTransferReport_		
Time Stamp	monthddyyyy	▼	HH-mm-ss
reate Unique File	<input type="checkbox"/>		
File Extension*	pdf ✓		

Configure: Web Form that will pass the parameter

In Adeptia Suite, go to Develop > Web Forms and click on Create New. Select 'Rich Form' and click on Create Form. Use Template option if Rich Form option is not granted in your license key. See next slide.

New Web Forms

Name*	<input type="text" value="WireTransferReportRequest"/>
Description*	<input type="text" value="Request form"/>
Application Type*	<input type="text" value="Form"/>
Form Type	<input type="text" value="Rich Form"/> <input type="button" value="Create Form"/>
Add as Link	<input checked="" type="checkbox"/>
MajorLevel Category Name	<input type="text" value="My Solutions"/>
MinorLevel Category Name	<input type="text" value="Wire Transfer"/>
Link Name	<input type="text" value="New"/>
Action	<input checked="" type="radio"/> Trigger Process <input type="radio"/> Customize
Select Process Flow	<input type="text" value="WireTransferFlow"/>
Upload File	<input type="text"/> -- Select Main File -- <input type="button" value="Add/Remove File(s)"/>

Configure: Web Form that will pass the parameter (contd.)

When the Form designer opens up change the default form theme from Workflow to Clear or Professional Blue in the Style tab as shown below. The purpose of doing this step is that this form would be available under Monitor > My Solutions and not as part of a run-time workflow task.

Info: -If you want to use this form as menu link then select a theme other than default theme(WorkFlowTaskTheme).Go to "Style(Tab

The screenshot shows the Form Designer interface. On the left is the 'PALETTE' containing various controls like Dropdown, Radio, Checkbox, Section, Repeat, Tabs, Panel, Message, Link, Trigger, Upload, Text, TextArea, Date, EMail, Money, Phone, Quantity, Number, Image, Video, and Submit. Below the palette is the 'CUSTOM' section and the 'PROPERTIES' window. The 'PROPERTIES' window has three tabs: 'Settings', 'Style', and 'Form Info'. The 'Style' tab is selected and has a red checkmark above it. It contains fields for Width (set to 'regular' and '600px'), Height, Controls (set to 'vertical'), Theme (set to 'Professional Blue' with a red checkmark), Font Name, Font Size, and Font Color.

The screenshot shows the 'WireTransferReportRequestForm' in preview mode. At the top, there are buttons for 'Preview', 'Controls', 'Rules', 'Form Action', 'Doc Action', 'Finish', and 'Cancel'. Below these is a 'print' button. In the center, there are three buttons: 'Save' (with a green checkmark), 'Reset' (with a red X), and 'Finish' (with a green checkmark). At the bottom, there is a text box with the instruction: 'Drop Submit buttons from the palette to add to the form.'

Configure: Web Form that will pass the parameter (contd.)

Click and drag Text from the Palette and put into the Form area. Make sure that the Field Name is same as the parameter name given in the Custom Report activity. Click Finish.

Info: -If you want to use this form as menu link then select a theme other than default theme(WorkflowTaskTheme).Go to "Style(Ta

The screenshot shows the software interface for configuring a web form. On the left, the 'PALETTE' contains various controls, with 'Text' selected. Below it, the 'CUSTOM PROPERTIES' panel shows the 'Name' field set to 'Account'. The main form area displays a 'Wire Transfer Request Form' with an 'Account' input field and a 'Finish' button highlighted with a red checkmark.

Value in the Name field should be exactly same as the one given in the Parameter name of the custom report activity (page # 26).

Configure: Web Form that will pass the parameter (contd.)

Click Save on the Web Form activity.

New Web Forms

Name*

Description*

Application Type* ▼

Form Type ▼

Add as Link

MajorLevel Category Name

MinorLevel Category Name

Link Name

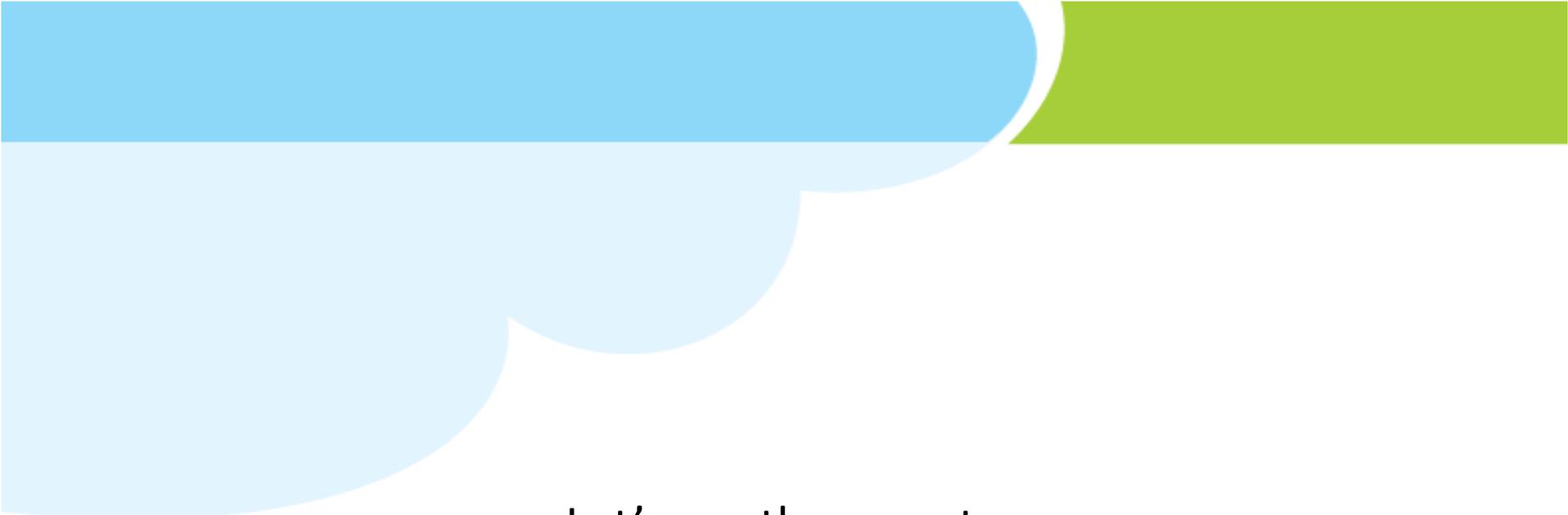
Action Trigger Process Customize

Select Process Flow ▼

Upload File -- Select Main File -- ▼

▶ Advanced Properties

* Mandatory fields.



Let's run the report

Monitor > My Solutions > Wire Transfer report

Enter a parameter value for Account and click on Finish.
Clicking on Finish will trigger the process flow and pass the Account value to the orchestration and the PDF report will be generated in the file target location.

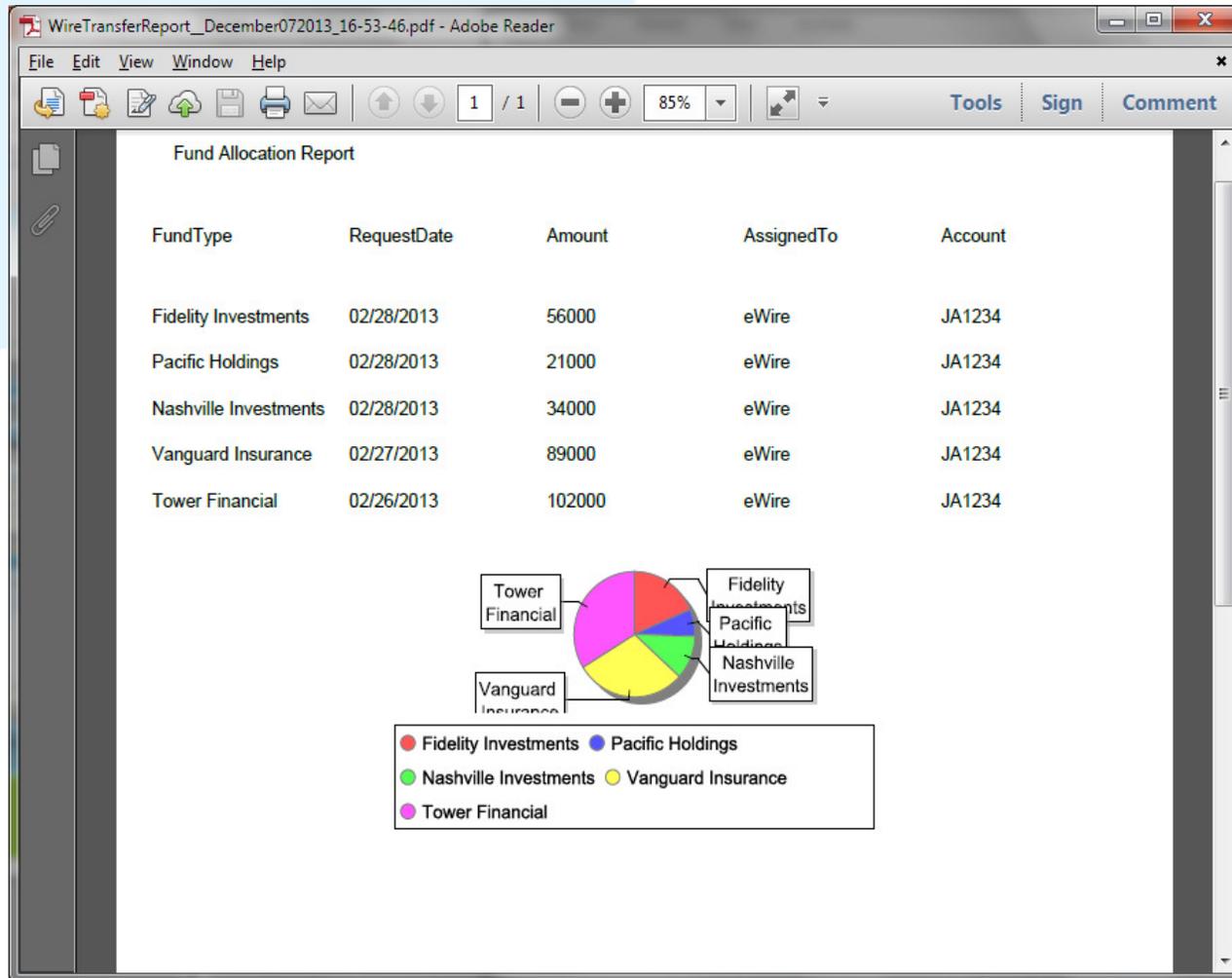
The screenshot displays the ADEPTIA web application interface. At the top, the ADEPTIA logo is visible. Below the logo is a navigation bar with tabs for 'Home', 'Develop', 'Monitor', and 'Administer'. The 'Monitor' tab is selected. Underneath the navigation bar, there are several menu items: 'My Tasks', 'My Documents', 'Dashboard', 'Reports', 'History', and 'My Solutions'. The 'My Solutions' menu item has a red checkmark next to it. Below the navigation bar, the breadcrumb trail reads 'My Solutions > Wire Transfer > New'. The main content area shows a 'Wire Transfer Request Form' with a 'print' icon and a 'Account' input field containing the value 'JA1234'. Below the input field are three buttons: 'Save', 'Reset', and 'Finish'. The 'Finish' button has a red checkmark next to it. To the right of the 'Finish' button is a blue arrow pointing to a message box that says 'Form is submitted and triggered the Process flow successfully'.

Look at the Report!

WireTransferReport_December072013_16-53-46.pdf

12/7/2013 4:53 PM

Adobe Acrobat D...



Manually executing a report

If the report is not generated by a run-time parameter or has a constant value in the Custom report activity's Parameter value then you can go directly to Monitor > Reports and click on the Execute icon next to the report name. This will open the report in a viewer.

Home Develop **Monitor** ✓ Administer

My Tasks ▾ My Documents ▾ Dashboard ▾ Reports ▾ ✓ History ▾ My Solutions ▾

Reports > Custom Reports

Quick Search

Action	Name	Description	Owner	Project Name	Modified
⚙️ ✓	WireTransferReport	wire transfer report	demouser	UseCase_Cust...	12/07/13 15:...
⚙️	StandardProcessflowReport	StandardPro...			07/26/06 13:...

Here we executed the standard process flow report to show you an example.

JasperViewer

Process Flow Execution Report

Report Between Sat, Dec 7, '13 12:00:00 AM and Sat, Dec 7, '13 11:59:59 PM

Generated by - Generated on: Sat, Dec 7, '13 05:08:47 PM

Process Flow: WireTransferFlow

Activity Name	Activity Type	Activity Status	Time (in Sec)
Executed at: Sat, Dec 7, '13 04:53:33 PM			
WireTransferReport	IndigoReport	Executed	11.24
ReportTarget	FileTarget	Executed	0.16
Time taken by transaction (in sec): 14.86			

Page 1 of 3

Manage Custom Report project

All the objects created for this use case can be managed by going to Develop > Projects

The screenshot shows the Adeptia web application interface. At the top, there is a navigation bar with the Adeptia logo and the text 'ADEPTIA'. To the right of the logo, there are user options: 'demouser' and 'Help'. Below the navigation bar, there are tabs for 'Home', 'Develop', 'Monitor', and 'Administer'. The 'Develop' tab is active, and it contains a sub-menu with 'Projects' (highlighted with a red checkmark), 'Solutions', 'Process', 'Services', 'Events', 'Web Forms', and 'Reports and Dashboards'. The main content area is titled 'Projects' and shows a list of activities for the project 'UseCase_Custom_Report'. The list is organized into categories: Custom Report (1), Database Info (1), File Target (1), Process Flow (1), and Web Forms (1). Each category contains one or more items with columns for Name, Description, Owner, Modified, and Action.

<input type="checkbox"/>	Name	Description	Owner	Modified	Action
Custom Report (1)					
<input type="checkbox"/>	WireTransferReport	wire transfer report	demouser	12/07/13 15:42	⋮
Database Info (1)					
<input type="checkbox"/>	WireTransferDB	db connection	demouser	12/07/13 15:41	⋮
File Target (1)					
<input type="checkbox"/>	ReportTarget	place report in folder	demouser	12/07/13 17:04	⋮
Process Flow (1)					
<input type="checkbox"/>	WireTransferFlow	flow to dynamically generate re...	demouser	12/07/13 16:49	⋮
Web Forms (1)					
<input type="checkbox"/>	WireTransferReportRequest	Request form	demouser	12/07/13 16:27	⋮

Leading the **Integration** Revolution

Your business problems have changed.
Why hasn't your integration solution?

Thank You!



ADEPTIA