

## Design Concepts of Adobe Interactive Forms Using Web Dynpro Java ( Adobe Live Cycle Designer 7.1 )

### Summary

This article gives a brief idea about the page designing in Adobe Live Cycle Designer 7.1. The Concepts are already mentioned in the Help, however article blog gives a simplified picture.

**Author:** Abhimanyu Lagishetti

**Company:** Covansys

**Created on:** 17 June 2007

### Author Bio



B.Tech Computer Science Graduate, In SAP since 2004. Worked in Various Technologies Webdynpro Java, Webdynpro ABAP, BSP, Interaction Center Web Client of CRM and ABAP.

## Table of Contents

Summary.....	1
Author Bio .....	1
Basic Knowledge Required.....	2
Master and Body Pages .....	2
Content Areas .....	2
Static and Dynamic Forms.....	2
Let's start with a simple Application ...	3
Place Property Determines where the subform will be laid down: .....	7
Rebuild, Deploy and Run the Application. ....	8
Disclaimer and Liability Notice.....	9

## Basic Knowledge Required

- Basic Knowledge on Web Dynpro Java and Adobe Interactive Forms.

## Master and Body Pages

- Body Pages are the actual pages of the Adobe Interactive Form.
- Every Interactive Form must have one Master Page, which is automatically generated.
- Master pages determine the size and orientation of body pages.
- By Default the body page inherits size and orientation from the master page.
- Each Body Page is created with a default subform that covers the whole page.

## Content Areas

- LiveCycle Designer creates default Content Area on the master page.
- You can add Content Areas to Master Pages Only.
- Content Areas define where objects can be placed on body pages. (Objects are the Library elements like button, static text etc...)
- You can not place Objects on a body page unless it is inside the area bounded by Content Area.

## Static and Dynamic Forms

- Static Form is the one where the content of the body is predetermined, where as in Dynamic Form the content of the form varies (Like in Tables).
- A static form design will typically contain one content area. A dynamic form design can have one or more content areas.

## Let's start with a simple Application ...

..., which demonstrates working with tables and dynamic content in Adobe Interactive Forms.

Create a Simple Web Dynpro Application

Create the Context as Follows

DataSource of cardinality 1..1, Customer of Cardinality 0..n

The screenshot shows the Adobe Live Cycle Designer interface. At the top, there are three tabs: 'CreateMedicalRef...', 'LocalDevelopmen...', and 'AIFTable'. Below the tabs is a window titled 'Context'. Inside this window, a tree view shows the following structure:

- Context
  - DataSource
    - Customer
      - FirstName
      - LastName
      - Nationality
      - Position
    - pdfSource

Below the tree view is a tabbed interface with tabs for 'Properties', 'Layout', 'Context', 'Plugs', 'Actions', 'Methods', and 'Implementation'. The 'Properties' tab is active, showing a table with the following data:

Property	Value
calculated	false
name	pdfSource
readOnly	false
structureElement	
type	binary

Fill the Customer Data up to 50 rows with test elements.

```

IPrivateAIFTableCompView.ICustomerElement customer = null;
for( int i=0; i<50;i++){
    customer = wdContext.createCustomerElement();
    customer.setFirstName("Test1");
    customer.setLastName("Test2");
    customer.setPosition("CEO");
    customer.setNationality("INDIAN");
    wdContext.nodeCustomer().addElement(customer);
}

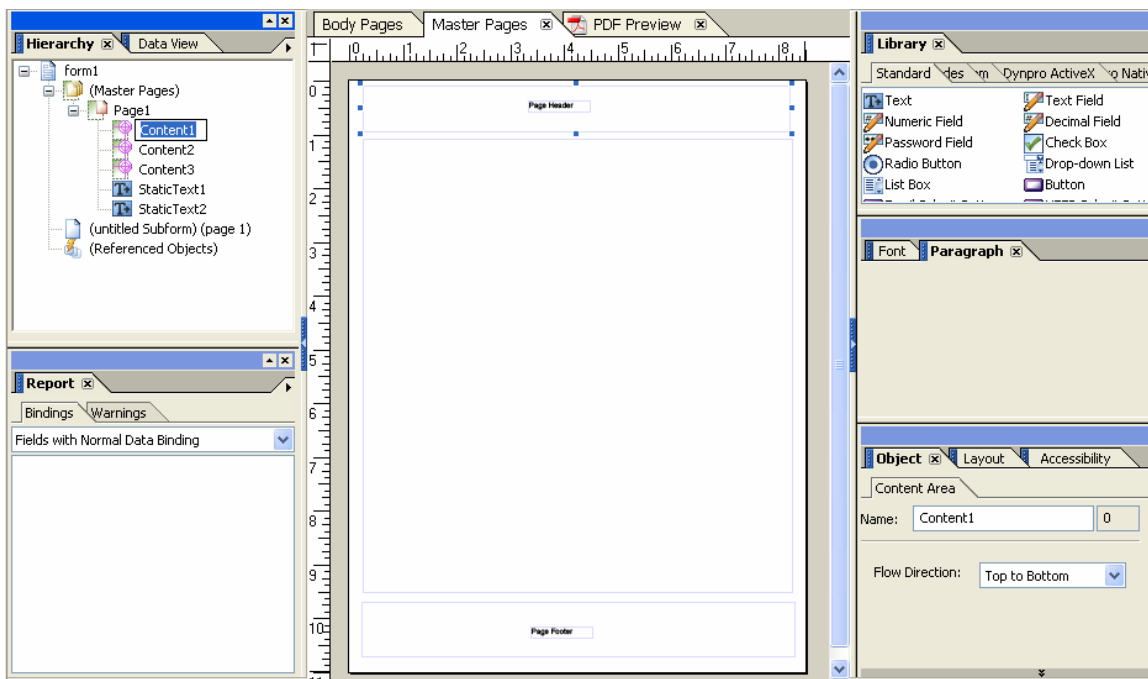
```

- Place Adobe Interactive Form UI Element in the View and change the Properties as follows:

Elementproperties of UIElement	
additionalArchives	
dataSource	DataSource
displayType	activeX
enabled	true
height	800px
id	InteractiveForm1
mode	updateDataInPdf
pdfSource	pdfSource
readOnly	false
templateSource	AIFTableCompView_InteractiveForm1.xdp
tooltip	<>
visible	visible
width	800px

•Right click on the Interactive form and choose Edit, Which Opens the Adobe Live Cycle Designer. Choose the Master Page tab, Observe it has only one Object which is Content Area. The Content Area defines the boundary of the page. You can place static content over Master Page.

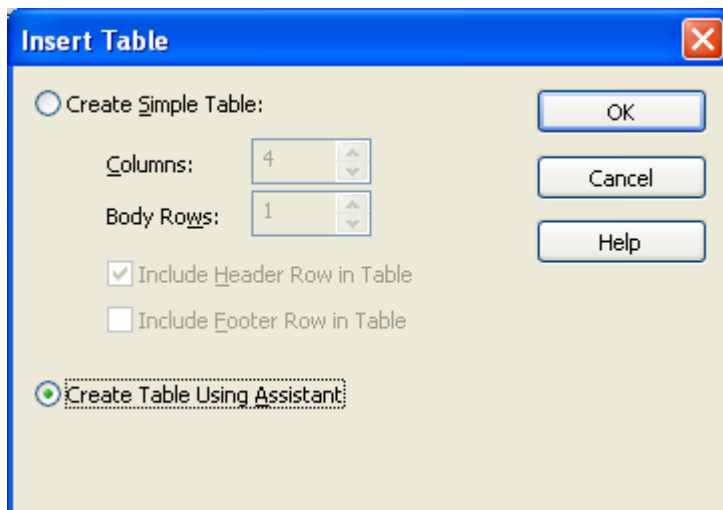
Place two more content Areas in Master Page, make sure there are no overlapping and place two static text elements to show as Header and Footer as follows



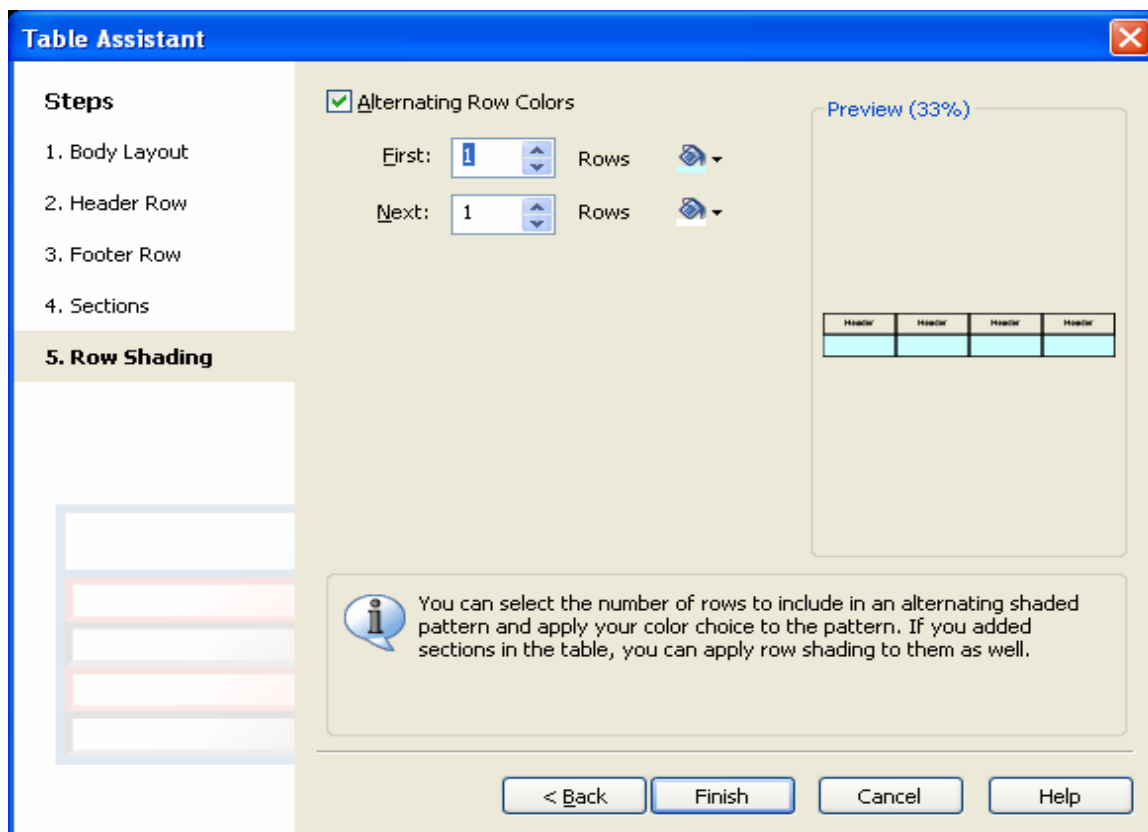
Choose Body Page Tab, Observe the default subform in the Body Page which is automatically created by the Designer.

Now Choose From the Menu table->insert table...

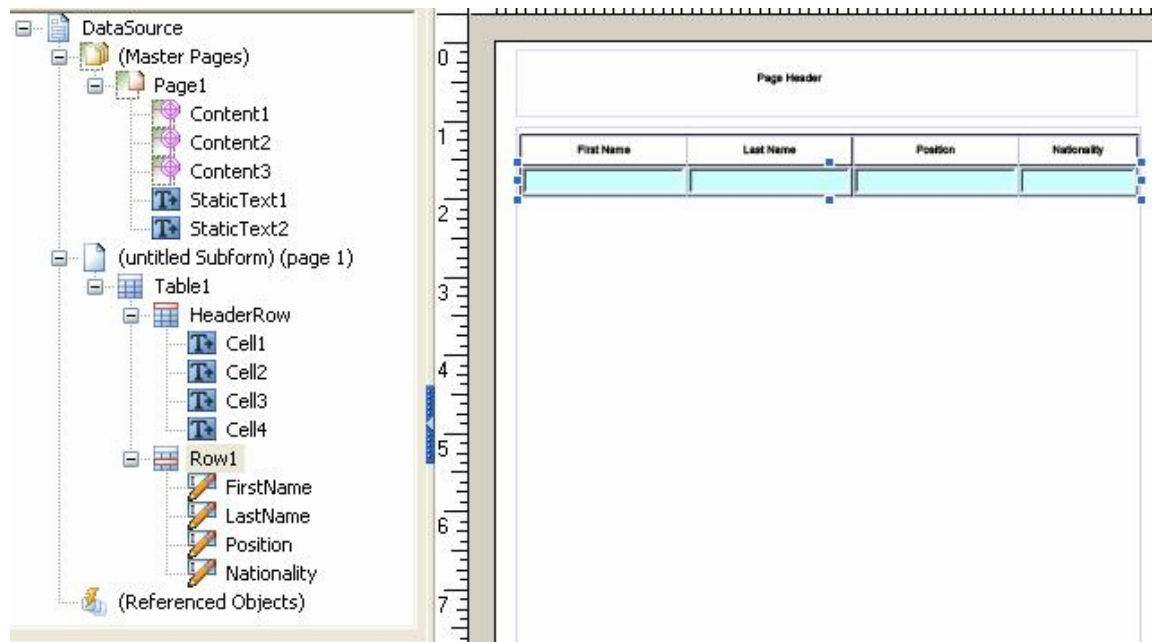
A dialog box appears choose the radio button create table using assistant( This Table Wizard is only from Designer 7.1, this feature is not there in earlier versions )



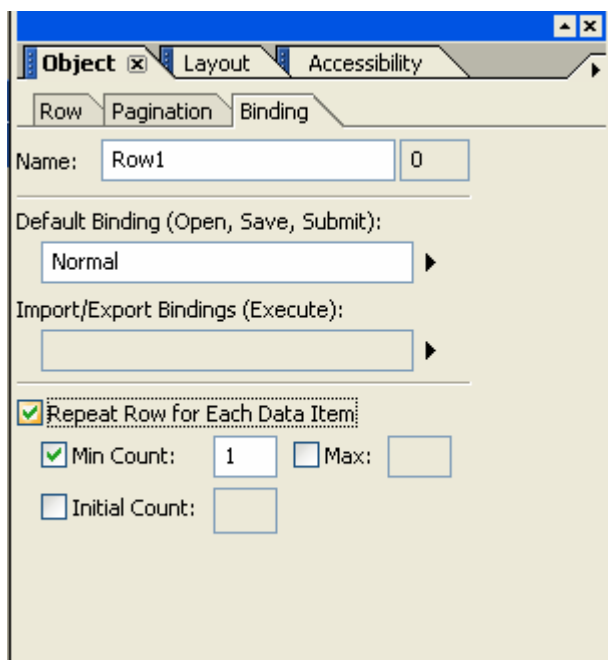
Choose the Number of Columns and Rows in the first step, choose the default values in the next steps and finish.



The Table Created goes into the Default Subform in the Body Page. Change the Headers and Bind the Cells with the Data View Context.



Select the Row1 Object and check the property Repeat Row for Each Data Item



Now Build your project, Deploy and Run

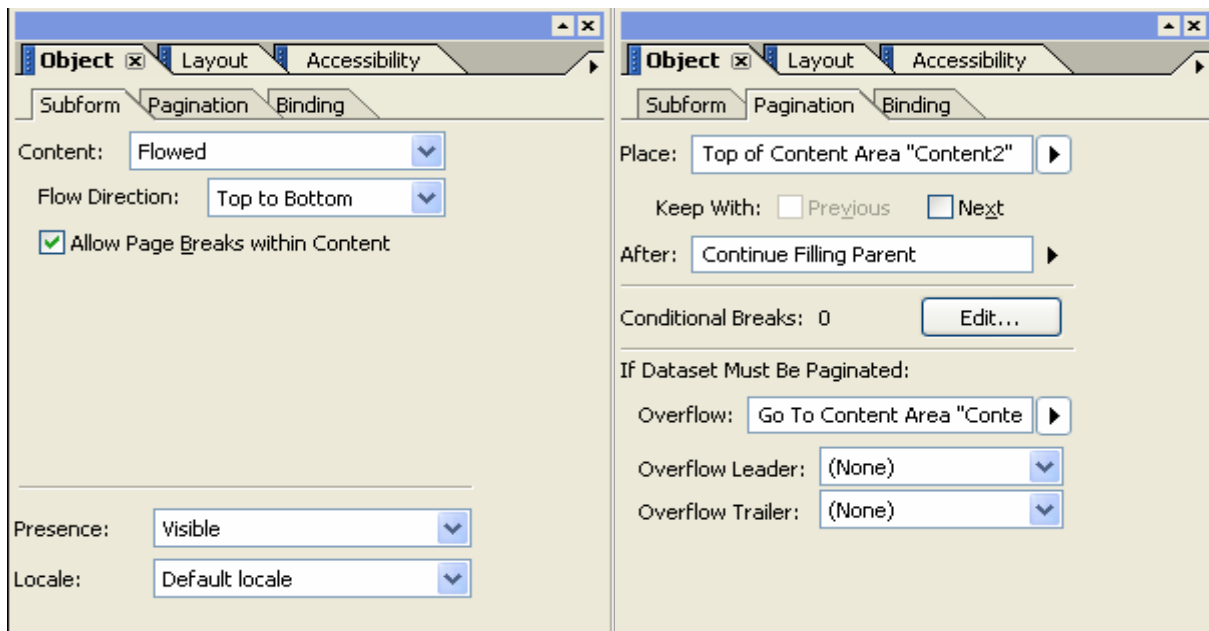
The Output is only in one page and the remaining records are left out.

To avoid this, Select the Subform and change the properties as follows:

In Object Palette choose tab Subform and specify the Content as 'Flowed'

In the same Palette choose tab Pagination and choose the Place as Top of Content

Area "Content2" and Overflow to Go To Content Area "Content2"



### Place Property Determines where the subform will be laid down:

- Following Previous Lays down the subform after the previous object in the parent subform.
- In Content Area > [name\_of \_content\_area] Lays down the subform in the specified content area.
- Top of Next Content Area Lays down the subform at the top of the next content area.
- Top of Content Area > [name\_of \_content\_area] Lays down the subform at the top of the specified content area.
- On Page > [name\_of \_page] Lays down the subform on the specified page (introduces a page break if one does not occur naturally).
- Top of Next Page Lays down the subform at the top of the next page (introduces a page break if one does not occur naturally).
- Top of Page > [name\_of \_page] Lays down the subform at the top of the page, whenever the specified page is rendered (introduces a page break if one does not occur naturally).
- Overflow Property Sets an overflow for a subform that spans pages.
- None Starts filling the next page (introduces a page break if one does not occur naturally).
- Go To Content Area > [name\_of \_content\_area] Starts filling the specified content area.
- Go To Page > [name\_of \_page] Starts filling the specified page.
- After Property Determines which area to fill after the subform has been laid down:
- Continue Filling Parent Continues merging data for all objects remaining to be filled in the parent subform.
- Go to Next Content Area Starts filling the next content area.
- Go To Content Area > [name\_of \_content\_area] Starts filling the specified content area.
- Go To Next Page Starts filling the next page.
- Go To Page > [name\_of \_page] Starts filling the specified page.

**Rebuild, Deploy and Run the Application.**

The Page Header and Page Bottom Appears and the Data is flowed in the Content area 2 Only without loss.



## **Disclaimer and Liability Notice**

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.