



HOW TO USE THE SERIALIZED DATA STRUCTURE IN K2 SMARTFORMS

2013-06-10



Contents

.....	0
Introduction.....	2
Example.....	2



INTRODUCTION

The example below will give an in depth explanation on how to use serialized data structures. For more information on the Serialize and Deserialize functionality in K2 connect for SAP, refer to [KB000712 – How to use the Serialize and Deserialize functionality for Data Structures in K2 connect for SAP.](#)

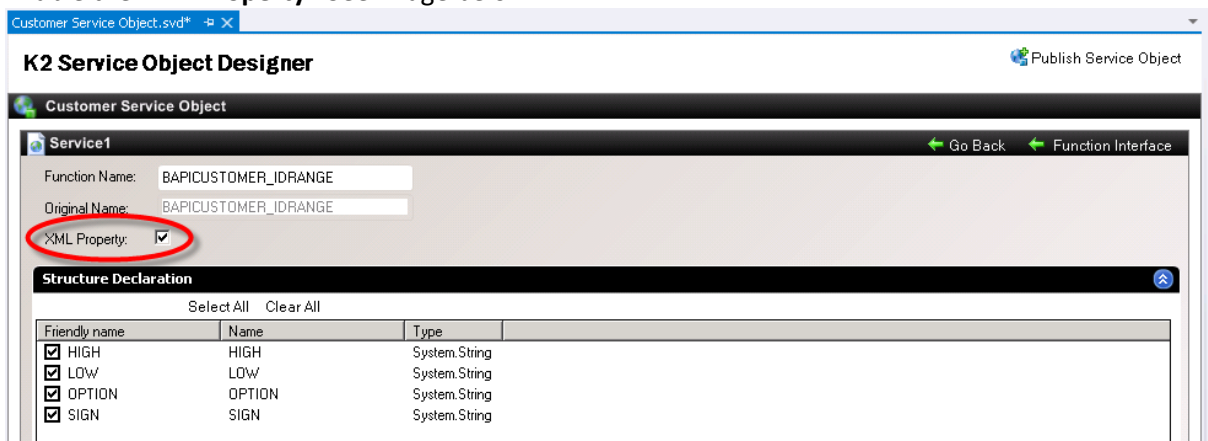
EXAMPLE

In SmartForms we will create a Form that will display the data returned from the Serialized Customer BAPI.

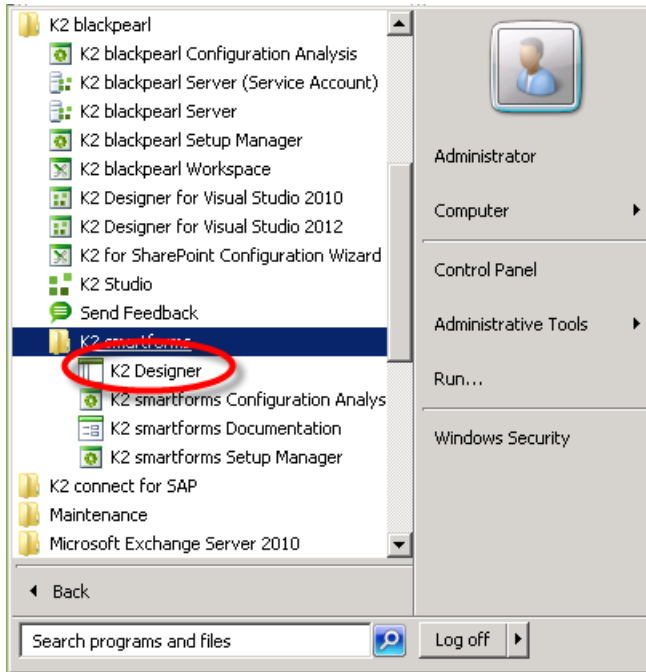
1. From K2 Designer for Visual Studio, create a new empty K2 project.
2. Add a K2 connect Service Object to the project.
3. Open the **K2 Service Object Designer** from the View menu.
4. Click on **Add Service**.
5. Add the **BAPI_CUSTOMER_GETLIST** filter to the BAPI Explorer.
6. Drag the filter to the new service as per the image below.



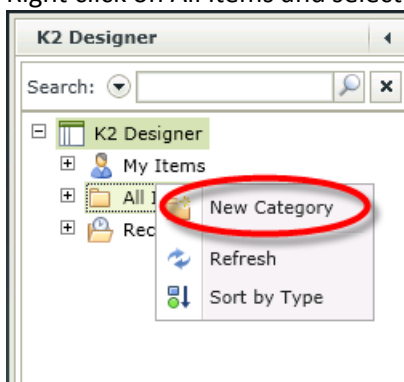
7. Click on the **Service Method** to open the **Method Designer**.
8. Click on the **IDRange** field and click on **Show Structure**.
9. Enable the **XML Property**. See image below.



10. Publish the Service Object.
11. We now need to create a **SmartObject** to interact with the **Form** that will be designed later on.
12. Open the SmartForms **K2 Designer** from the **Start** menu.

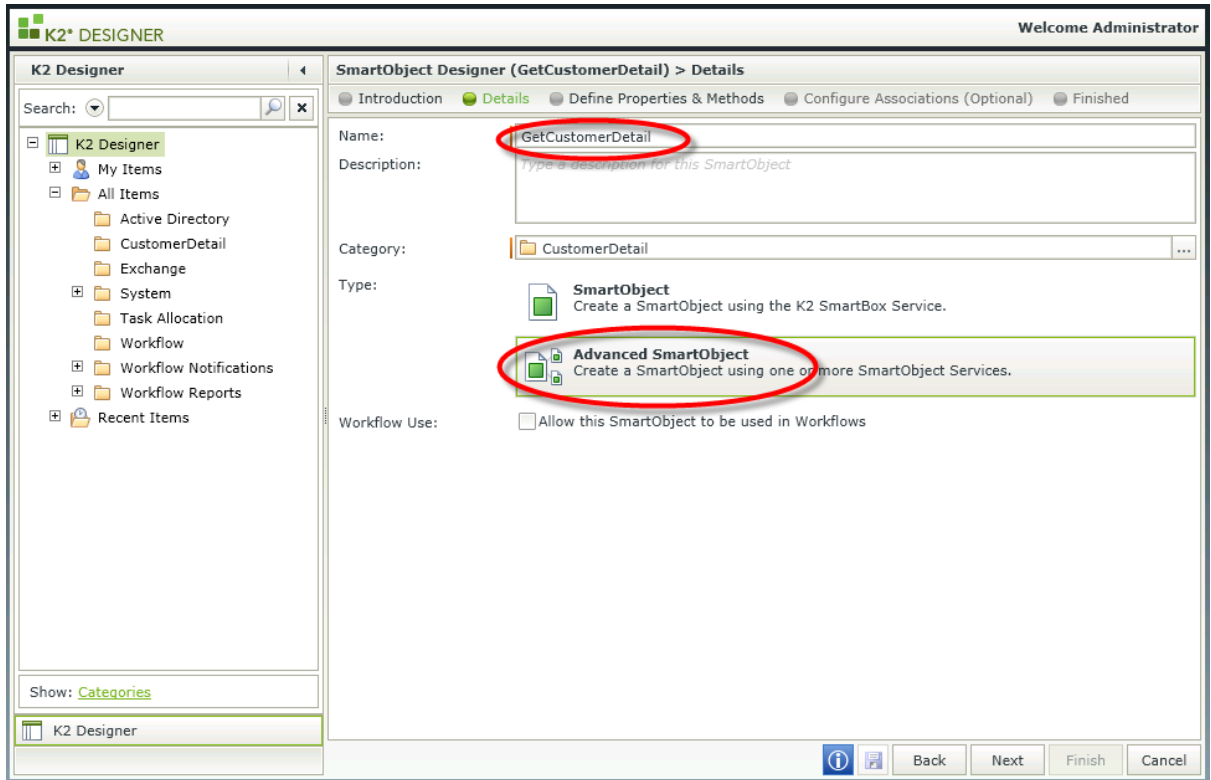


13. Right click on All Items and select **New Category**.

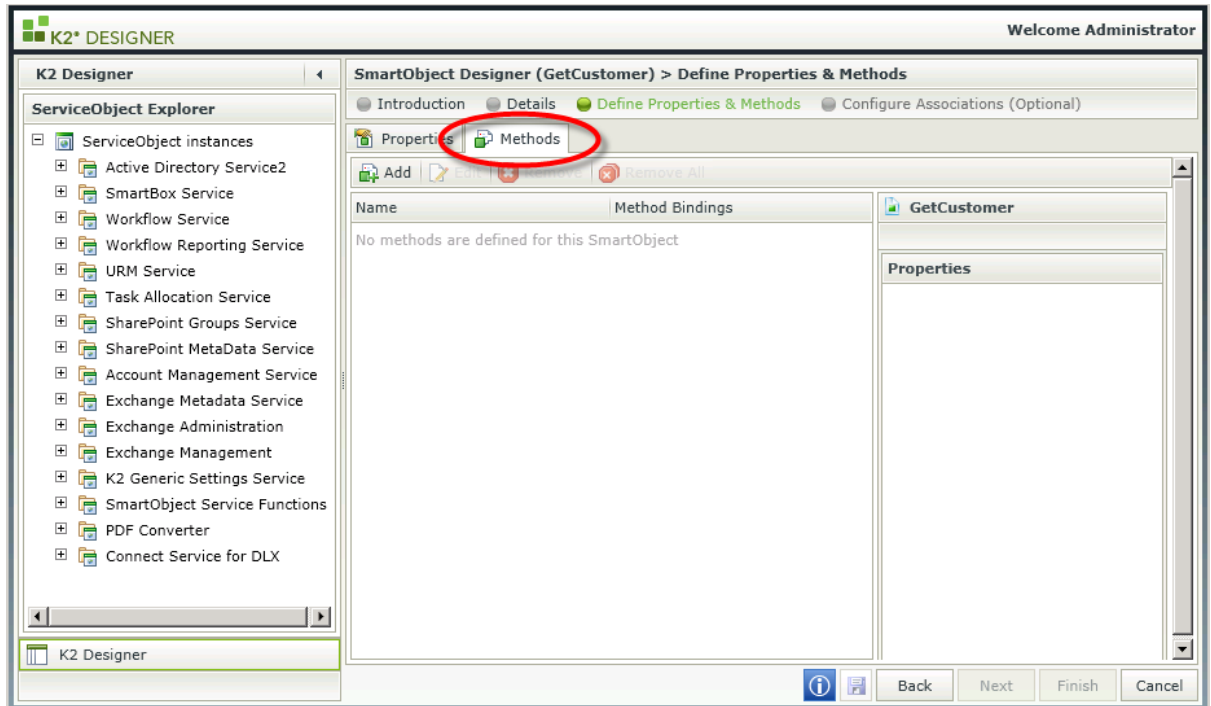


14. Name the Category **CustomerDetail**.

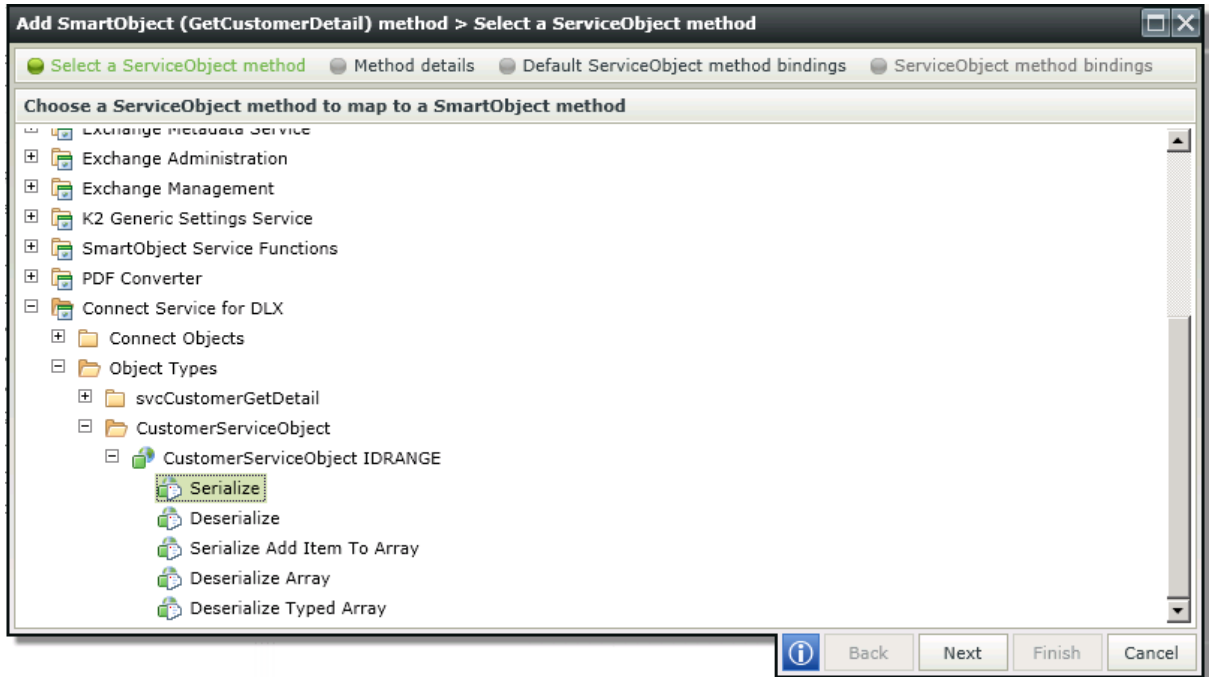
15. Right click on the new category and select **New SmartObject**. Name the SmartObject **GetCustomerDetail**. Select the **Advanced SmartObject** option.



16. Click Next.
17. Add a Property of type **Memo** to map to the **IDRange** Service Object property further down in the steps. Name the Property **CustomerSerializedItem**.
18. Select the **Methods** tab and click on Add.

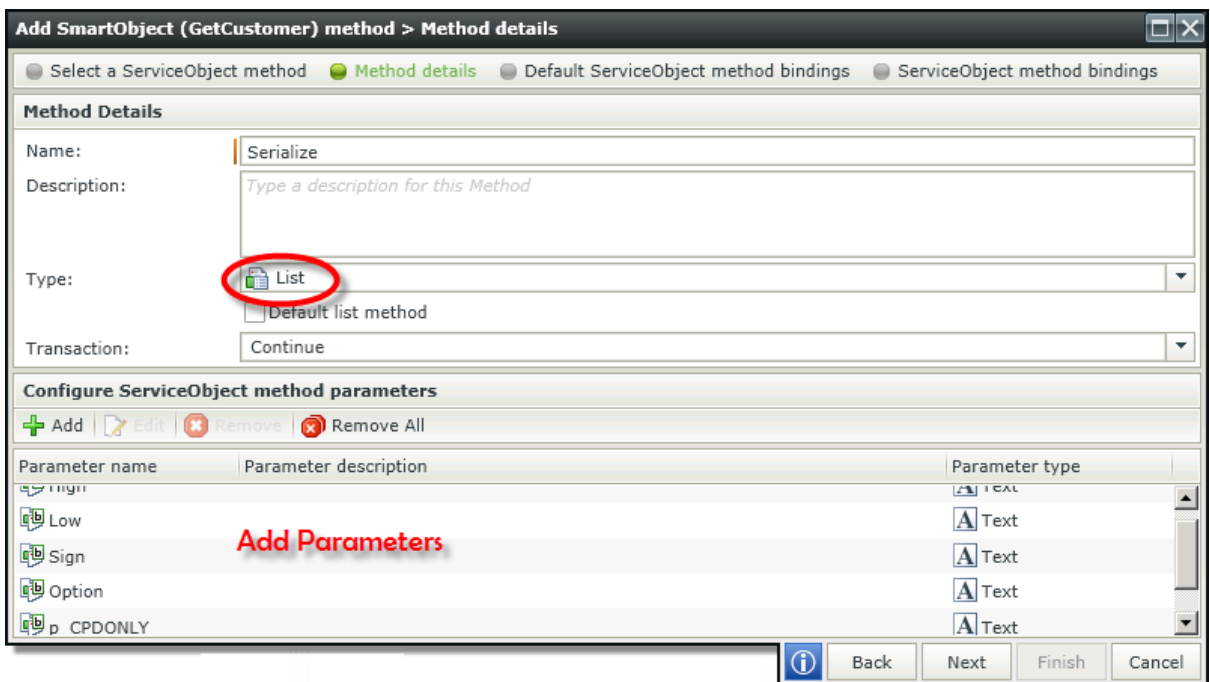


19. We need to add methods to build a serialized object to be used in the Form.
20. Add the Serialize method by expanding the Connect Service down to the Service Object previously created.



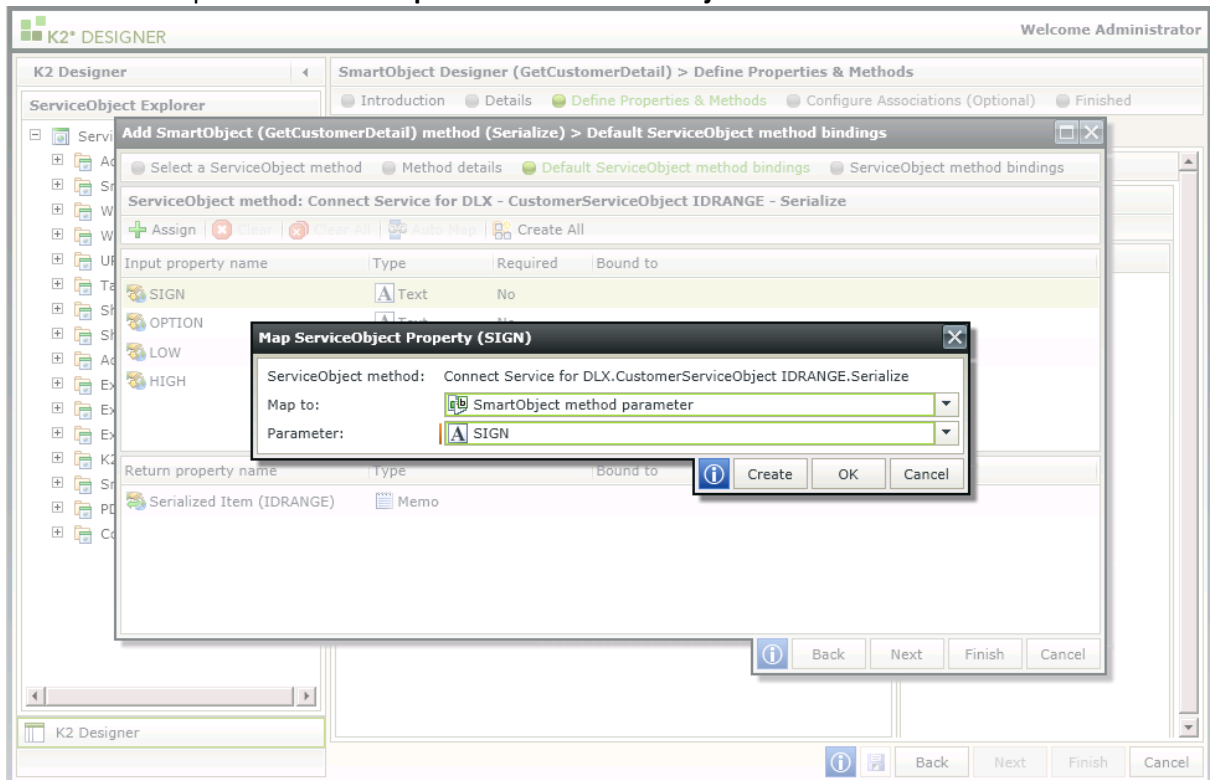
21. Click Next.
22. Select type List for the Method Details and add the following parameters. These parameters are used to build the serialized object.

Parameter Name	Parameter Type
HIGH	Text
LOW	Text
SIGN	Text
OPTION	Text
p_CPDONLY	Text
p_MAXROWS	Number

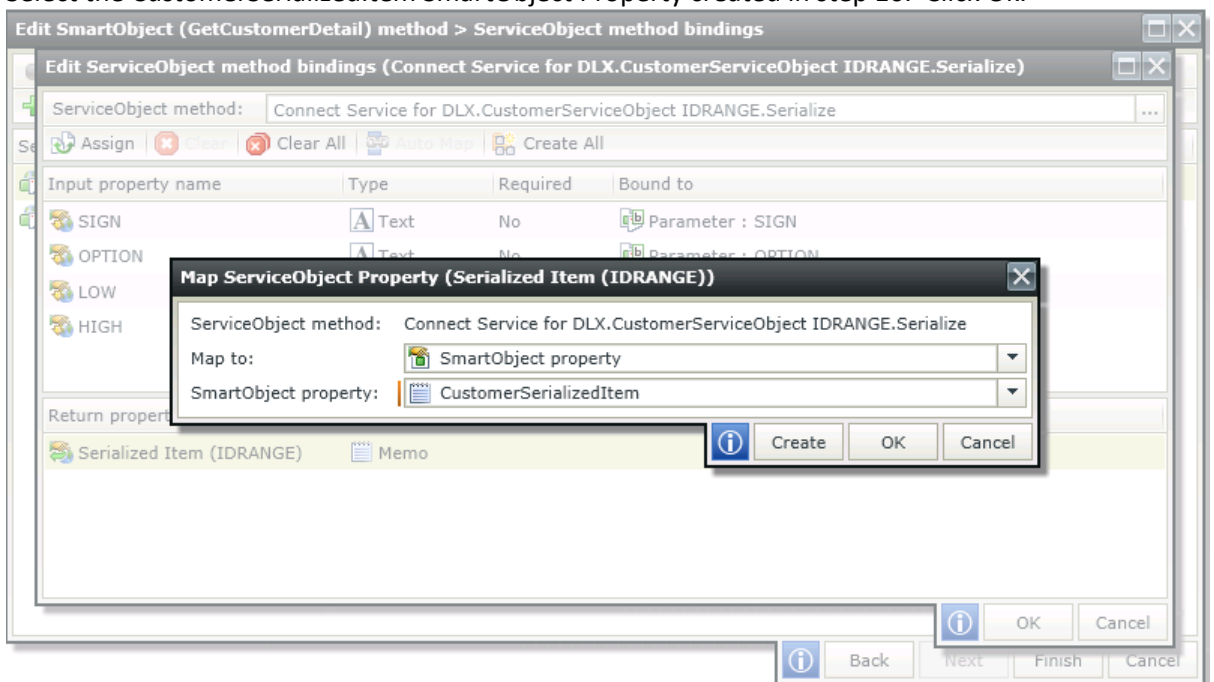




23. Click Next.
24. Select the **SIGN** input property and click on **Assign**.
25. Click on the dropdown next to **Map to** and select **SmartObject Method Parameter**.



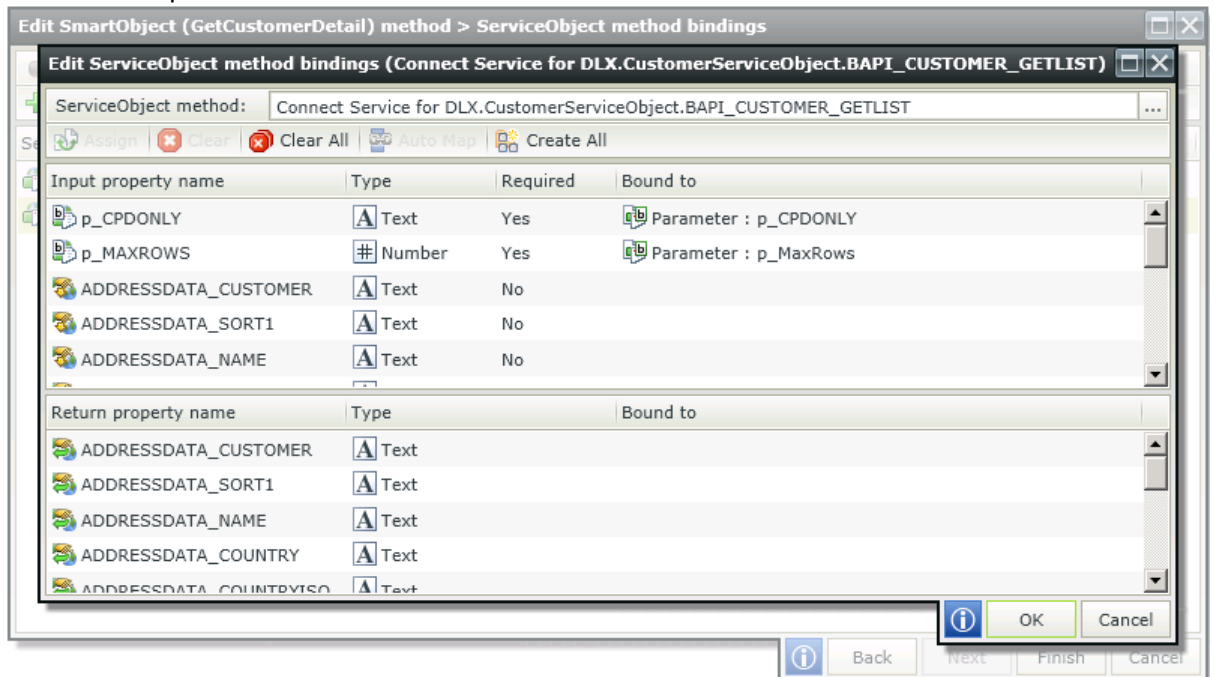
26. Click on the dropdown next to **Parameter** and select the **SIGN** parameter created in step 19.
27. Click OK.
28. Do the same for all the Input Properties.
29. Select the **Return Property** and click **Assign**.
30. Select the CustomerSerializedItem SmartObject Property created in step 16. Click Ok.



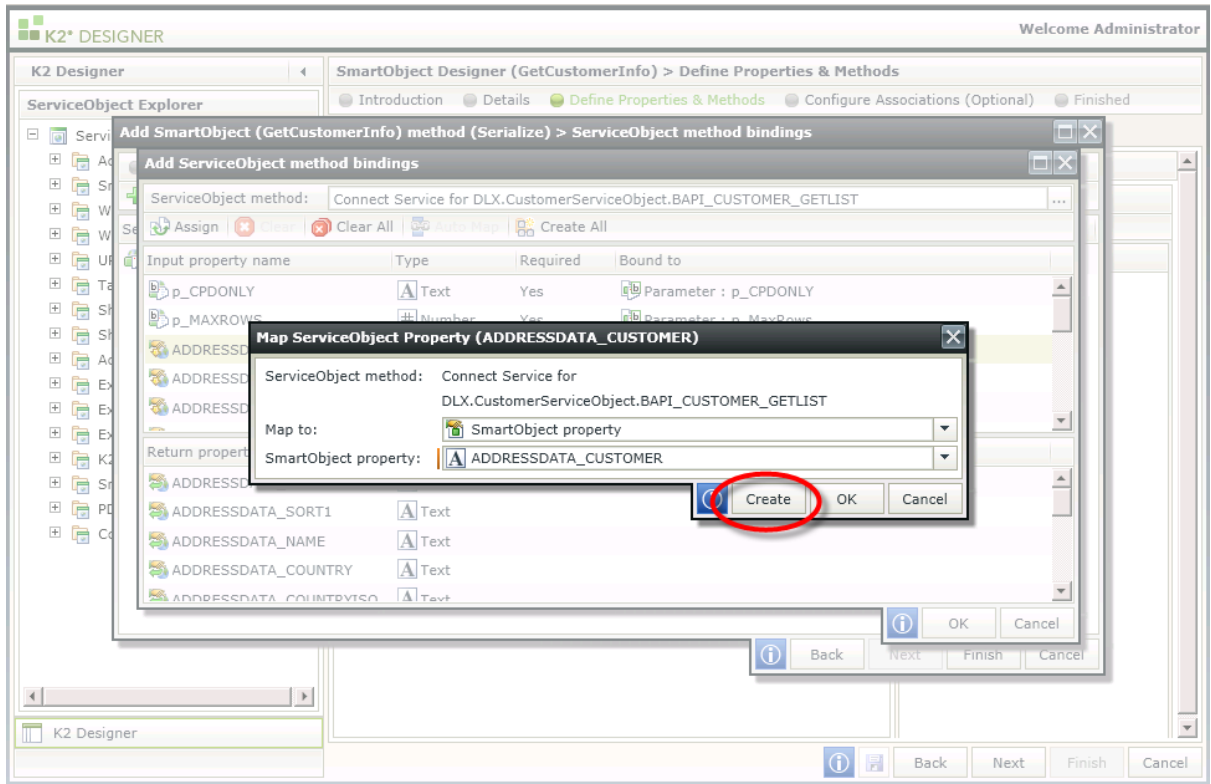
31. Click Next.



32. We now need to configure the BAPI_CUSTOMER_GETLIST Service Object method to return the serialized data to the Form.
33. Click Add to add **Service Object Method Bindings**.
34. Click on the ellipses next to **Service Object Method**.
35. Expand the **Connect Service** node.
36. Expand the **Connect Objects** node down to the **BAPI_CUSTOMER_GETLIST** Service Object method. Select the method and click OK.
37. From the Input Properties, scroll down to the **IDRange** Property and click Assign.
38. Select the SmartObject property created in step 16 to map to the **IDRange** Input Property.
39. Assign the two required Input Properties to the related SmartObject method parameters created in step 20.



40. Assign the Input and Return Properties you require by creating new SmartObject properties for each.

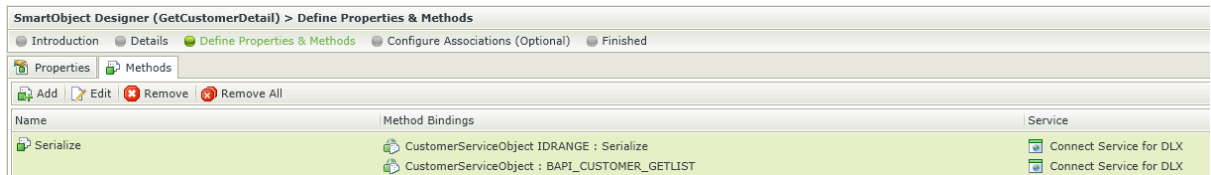


41. For this example we will only create properties for the following Input and Return properties:

- ADDRESSDATA_CUSTOMER
- ADDRESSDATA_NAME
- ADDRESSDATA_COUNTRY
- ADDRESSDATA_CITY
- ADDRESSDATA_POSTL_COD1
- ADDRESSDATA_TEL1_NUMBER

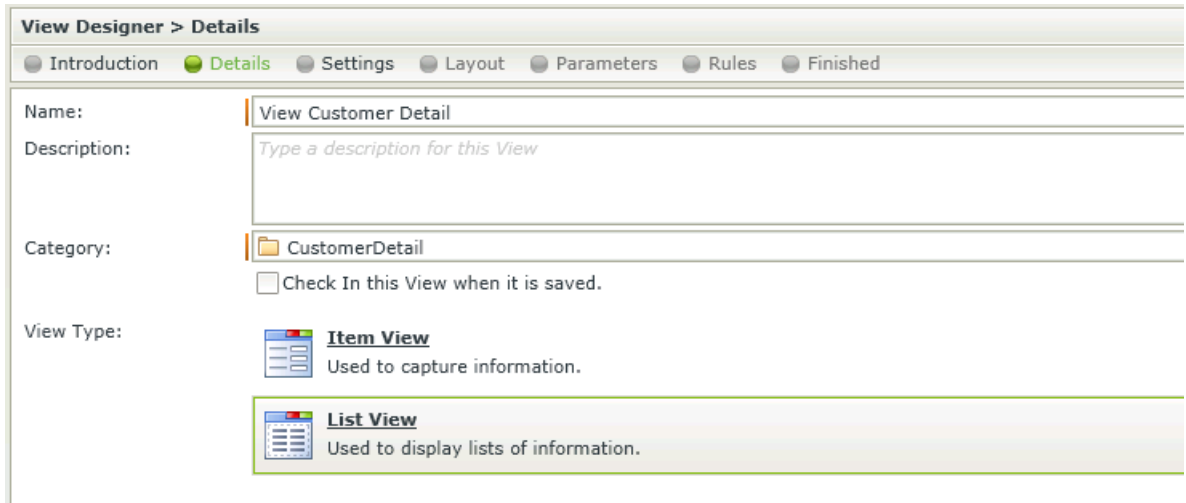
42. Click OK and click Finish.

43. The SmartObject method must be the same as the image below. This method will now build a serialized object for the Form controls that will be added further down in this example. The serialize data will then be returned when executing the Form. This is all configured on the Serialized SmartObject method and will allow you to only use 1 method when designing the Form.

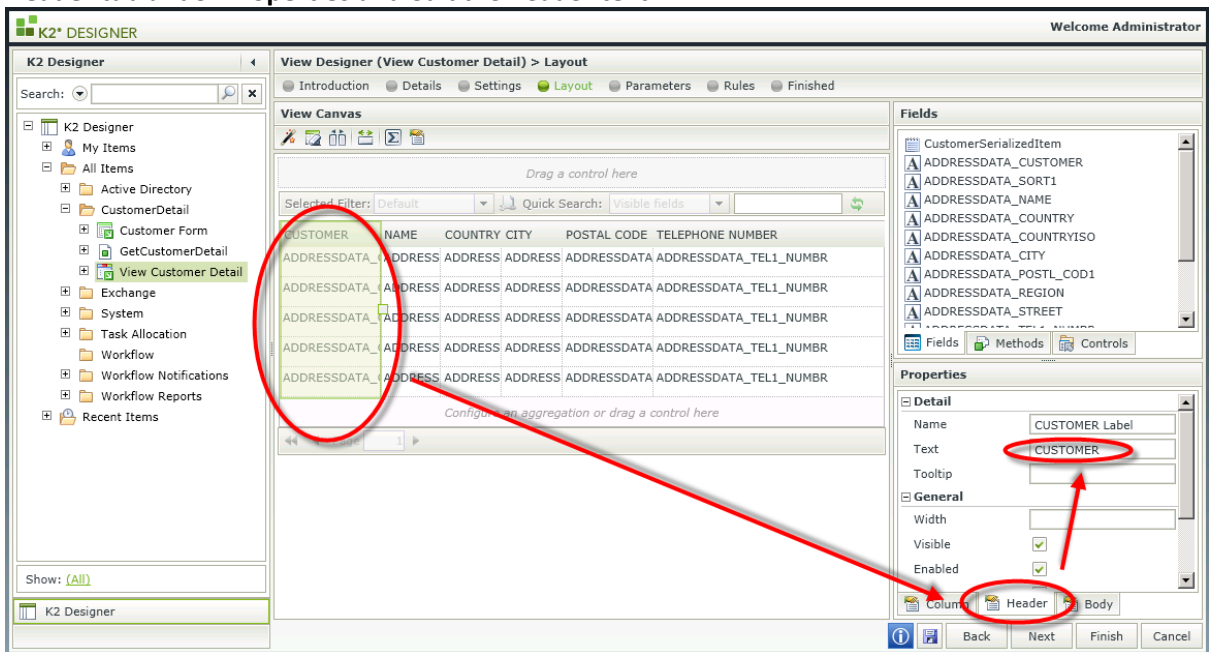


44. Click Finish.

45. Right click on the **GetCustomerDetail** SmartObject from the Category tree and select **Design a View**. Selecting the View to be a **List View**.

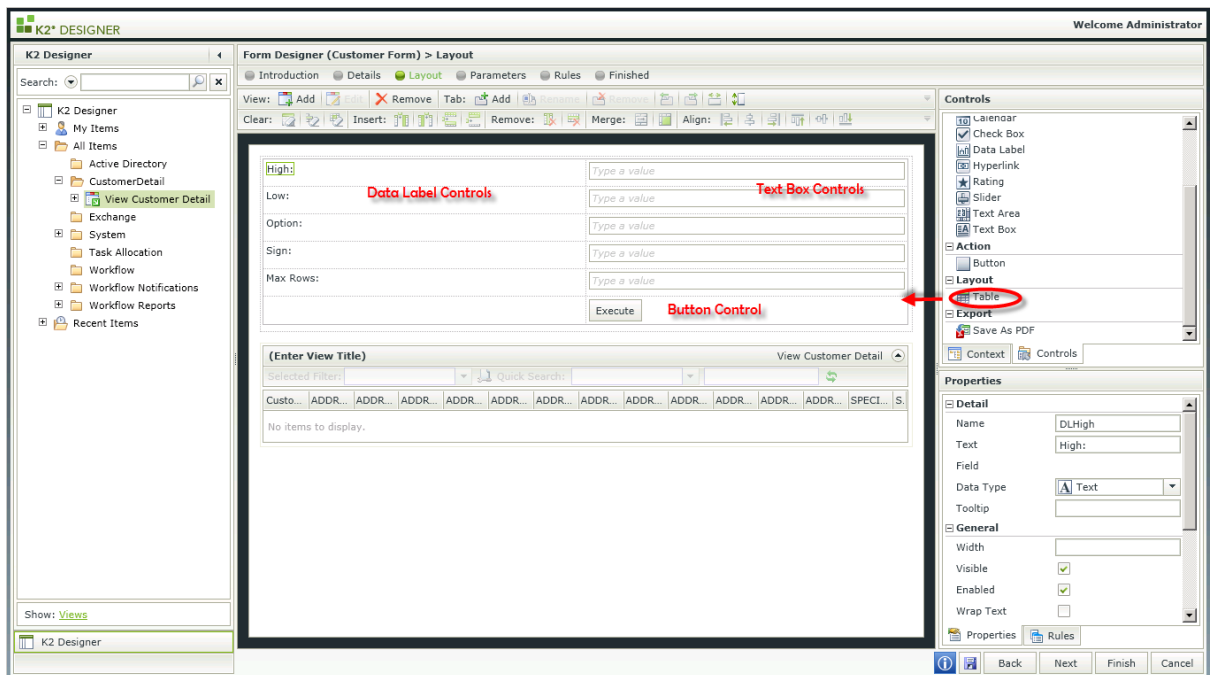


46. Click Next.
47. Select Generate View from the selected Data Source and include all fields. The fields are the SmartObject properties.
48. Click Next.
49. On the **Layout** you can change the column heading names by selecting a column, click on the **Header** tab under **Properties** and edit the header **text**.

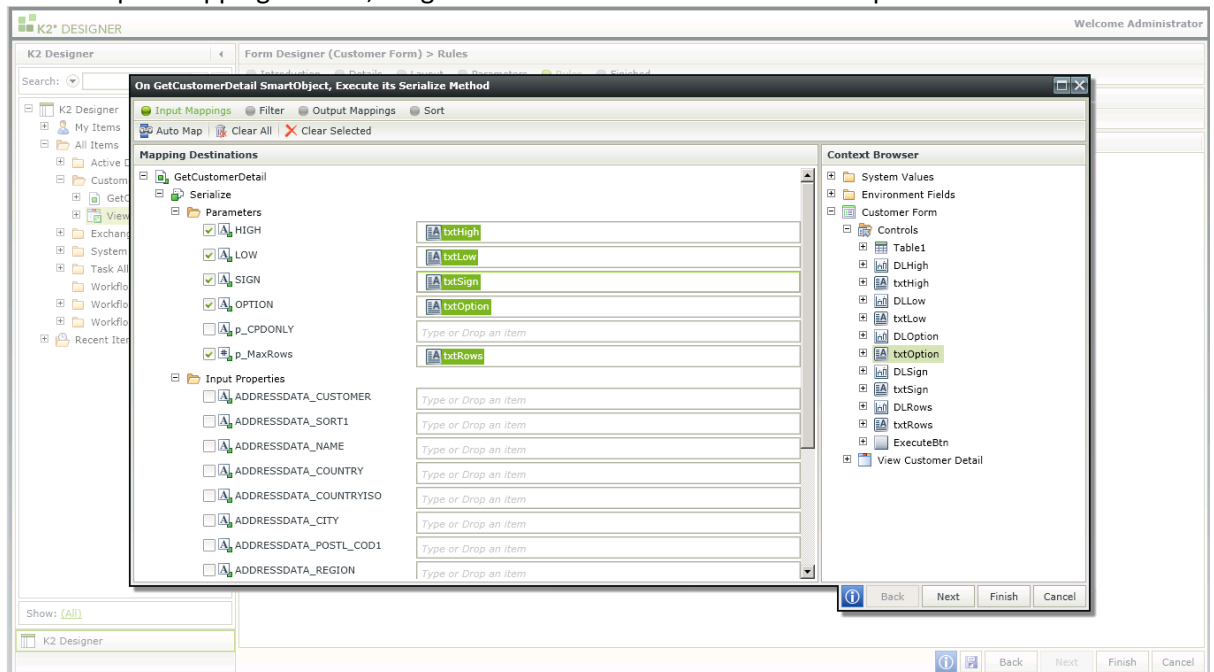


50. Click Finish.
51. Right click on the View from the Category tree and select **Design Form**.
52. Click Next.
53. Name the Form **Customer Form** and click Next.
54. Drag a **Table** control above the existing **List View** on the Form.
55. Configure the table with 2 columns and 6 rows.
56. Drag a **Data Label** control in the first 5 rows of column one. Rename the data labels.
57. Drag a **Text Box** control in the first 5 rows of the second column.
58. Drag a **Button** control to the last row of the second column and rename the button to **Execute**.

HOW TO USE THE SERIALIZED DATA STRUCTURE IN K2 SMARTFORMS

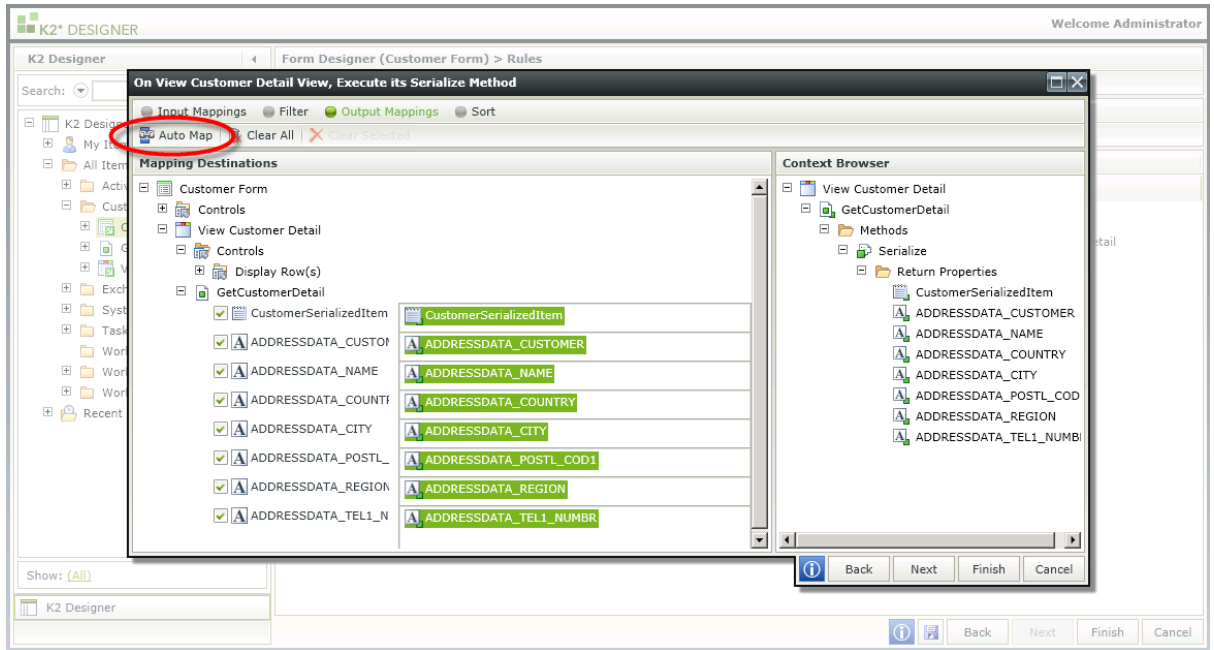


59. Click Next and Next again to go to the Rules section.
60. Add a new rule.
61. Select the **When a control on the Form raises an event** from Events.
62. Select the **Execute** button as the control in the rule definition and ensure the event is set to **Clicked**.
63. Select **Actions** and add the **Execute a View Method** action.
64. In the rule definition select the **View** to be executed and select the **Serialize** method.
65. Click on **configure**.
66. On the Input Mappings screen, drag the textbox controls to the relevant parameters.

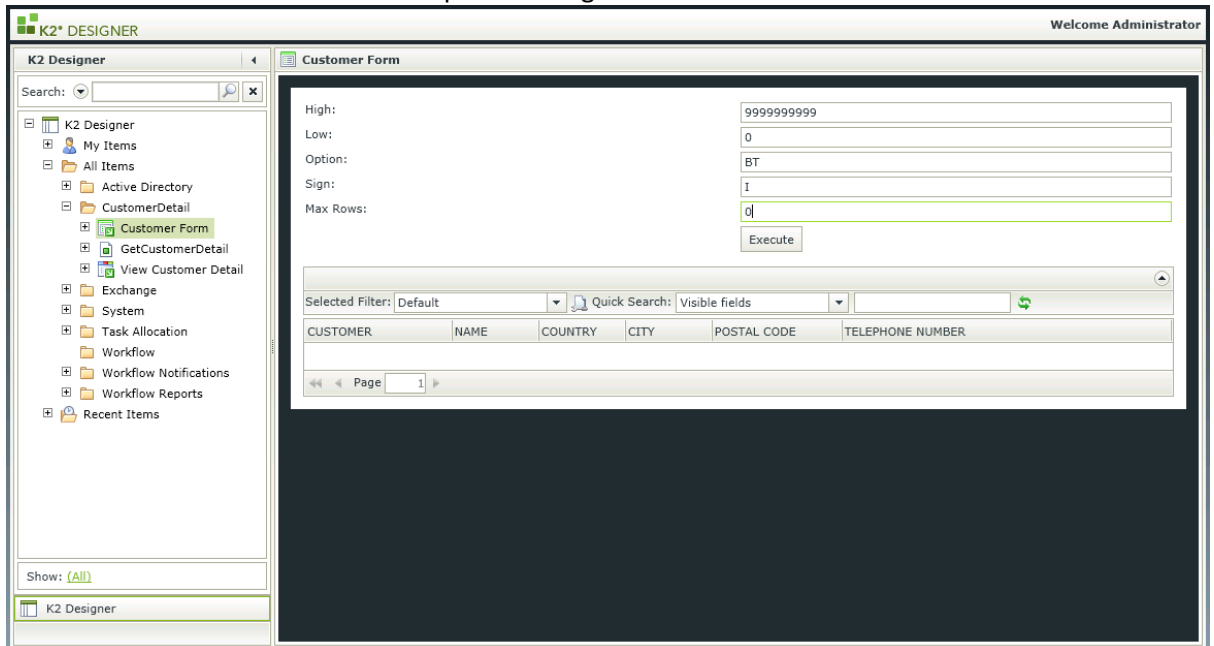


67. Click on the Output Mappings and click on **Auto Map**.

HOW TO USE THE SERIALIZED DATA STRUCTURE IN K2 SMARTFORMS



68. Click Finish.
69. Click Finish on the rule.
70. Finish the Form.
71. Right click on the Form from the Category tree and select **Run**.
72. Enter the values in the text boxes as per the image below.



73. Click on **Execute**.
74. When clicking on Execute, the data from the Customer BAPI is built as a Serialized Object. This allows the data to return as seen below.

HOW TO USE THE SERIALIZED DATA STRUCTURE IN K2 SMARTFORMS



The screenshot shows the K2 Designer interface. On the left is a tree view of the project structure, including folders like 'My Items', 'All Items', 'Active Directory', 'CustomerDetail', 'Customer Form', 'GetCustomerDetail', 'View Customer Detail', 'Exchange', 'System', 'Task Allocation', 'Workflow', 'Workflow Notifications', 'Workflow Reports', and 'Recent Items'. The main area displays a 'Customer Form' with fields for 'High', 'Low', 'Option', 'Sign', and 'Max Rows', each with a corresponding input field. Below the form is an 'Execute' button. Underneath the form is a data table with columns: CUSTOMER, NAME, COUNTRY, CITY, POSTAL CODE, and TELEPHONE NUMBER. The table contains 10 rows of customer data. At the bottom of the table is a pagination control showing 'Page 1'.

CUSTOMER	NAME	COUNTRY	CITY	POSTAL CODE	TELEPHONE NUMBER
0000000001	Nelson Tax ...	US	Philadelphia	19115	(215) 665 4534
0000000002	Wett	DE	Walldorf	69190	
0000000099	Einmalkunde	DE			
0000000110	Auto Klement	DE	München	81737	089/93534
0000000224	Reference fo...	US	Los Angeles	90025	
0000000255	Emma Bull	US	Denver	80216	303-297-4568
0000000256	Kenny A Ch...	US	Boulder	80303	3034493800
0000000257	John Evans	US	Nederland	80466	3032889463
0000000258	Roger Zahn	US	Albuquerque	87123	505-292-7426
0000000259	Laurel K. Ha...	US	Greely	80631	9703738444