



AltioLive Developer Training

Version 5.4

8. Using the Prototyping Wizard

Summary

This optional module describes how to use the AltioLive Prototyping Wizard.

Integra SP

88 Wood Street London
EC2V 7RS
United Kingdom

www.altio.com

tel: +44 (0) 20 8528 1045

Contents

The Prototyping Wizard	2
Overview of the Prototyping Wizard	2
Building an Application with the Prototyping Wizard	2
Defining the data and build the framework with the Prototyping Wizard.....	3
See the generated framework with the Designer	5
Creating Relationships with the Prototyping wizard	7

The Prototyping Wizard

The Prototyping Wizard provides an easy way to create the basic components of an application. Using the **Altio DB Manager** as the Back end application, the Wizard can be a useful tool for RAD/JAD prototyping activities.

This session assumes that you are familiar with the AltioLive Developer environment: the Presentation Server Administration, the Application Manager, and the Designer.

During the session we will look at:

- An overview of the Prototyping Wizard
- How to start building an application prototype using the Prototyping Wizard
- How to build a Master-Detail relationship between two lists

Overview of the Prototyping Wizard

The Prototyping Wizard ties together and automates the three main steps to creating a prototype application:

1. The definition of the (Altio DB) XML database structure and sample data.
In this step the elements and attributes of the application database are defined. The format of the XML data is relational: Master-Detail relationships between the elements can be defined.
2. The generation of Service Functions and Datapool definitions.
All the Service Functions and Datapools are created automatically, enabling records to be fetched, added, updated, and deleted from the Altio DB database.
3. Application design and view definitions.
The basic 'views' of the data are created automatically. Windows are generated, they list the sample data, and allow data to be added, edited and deleted.

Building an Application with the Prototyping Wizard

We are going to use the Wizard to create a standard application. The application comprises a database of departments and a database of employees.

A sample of the data follows:

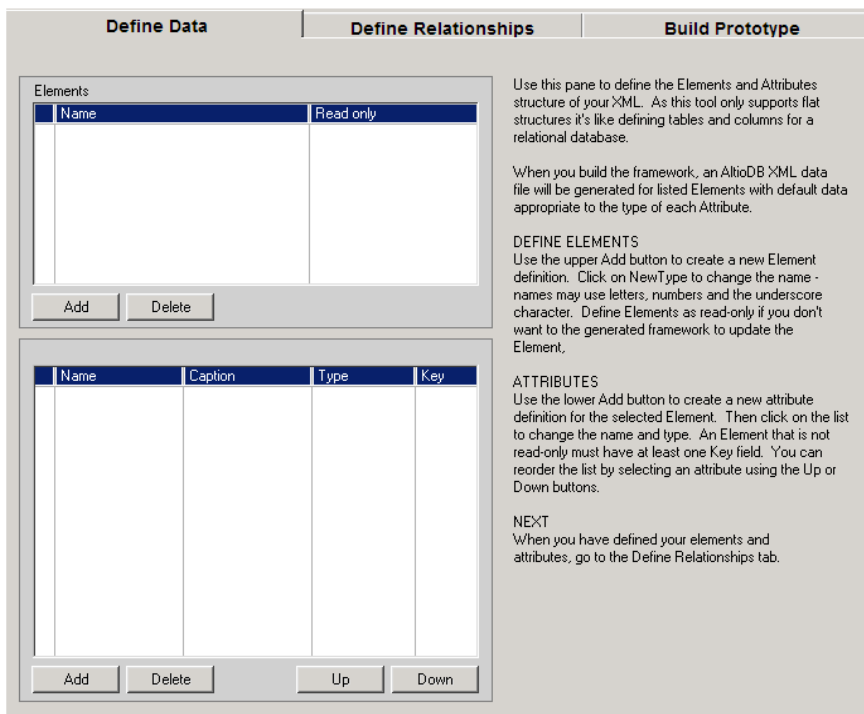
```
<names_CONTAINER>
  <names name_id='name_id1' email='email1'
    department_id='department_id1' last='last1' first='first1'/>
  <names name_id='name_id2' email='email2' department_id='department_id2'
    last='last2' first='first2'/>
  <names...
</names_CONTAINER>
<departments_CONTAINER>
  <departments department_id='department_id1' department_name='departement_name1'/>
  <departments department_id='department_id2' department_name=' departement_name2'/>
  <departments...
</departments_CONTAINER>
```

Defining the data and build the framework with the Prototyping Wizard

In this practical we will define the data and default views using the Wizard. We will open the Designer and review the generated application.

1. From AltioLive Studio, create a new project give it a name, for example: **departments**
2. From the Studio Explorer, right-click the **departments** project, and select the **Prototyping Wizard** tool from the context menu.

The Wizard displays:



We are going to define the data and then build the Framework.

Usually we would define the relationships between elements at this time. However, we will return to the Wizard to explore how a framework can be refined further.

3. Make sure the **Define Data** tab is active.
4. Add two new elements by clicking the **Add** button below the **Elements** list.
5. Name them **names** and **departments**:



6. The **Read only** check boxes should be left unchecked.

Please note: checking the Read only check boxes will result in only GET service requests being created for the data and no Datapools.

7. Select the **names** element.

8. Add five new attributes by clicking the **Add** button below the **Attributes for names** list:

Attributes for <names>				
Name	Caption	Type	Key	
name_id	ID	Integer	<input checked="" type="checkbox"/>	
department_id	Dept ID	String	<input type="checkbox"/>	
first	Firstname	String	<input type="checkbox"/>	
last	Lastname	String	<input type="checkbox"/>	
email	Email	String	<input type="checkbox"/>	

9. Select the **department's** element.

10. Add two new attributes by clicking the **Add** button below the **Attributes for departments** list:

Attributes for <departments>				
Name	Caption	Type	Key	
department_id	Dept_ID	String	<input checked="" type="checkbox"/>	
department_name	Department	String	<input type="checkbox"/>	

We have now defined the XML structure.

Please note: the Wizard will provide the containers for the elements: **<names_CONTAINER>** and **<departments_CONTAINER>**.

11. Click the **Build Prototype** tab.

Please note: There are two choices here:

- **Build:** builds the framework;
- **Advanced Options:** allows selection of the various parts of the build process.

12. Click the **Build** button:

The Wizard will show the progress of the build. When completed the screen will appear as follows:

Build	Advanced Options
-------	------------------

Results of build run at 16:00:04 on 04-Aug-2008:

SE Admin:
Registered application:
Using app directory:
C:\Documents and Settings\documentation\Desktop\July Altio
Version\omcat\webapps\altio52\WEB-INF\classes\apps\DPs/

AltioDB:
Created database file(s):
names.xml departments.xml
Linked XML data files into AltioDB

App Manager:
Created Service Function(s):
GET_names NEW_names UPDATE_names DELETE_names
GET_departments NEW_departments UPDATE_departments DELETE_departments

Created Datapool(s):
namesPOOL departmentsPOOL

Created Datakey(s):
names departments

Designer:
Created data type definitions
Created view definition DPS_view


FINISHED

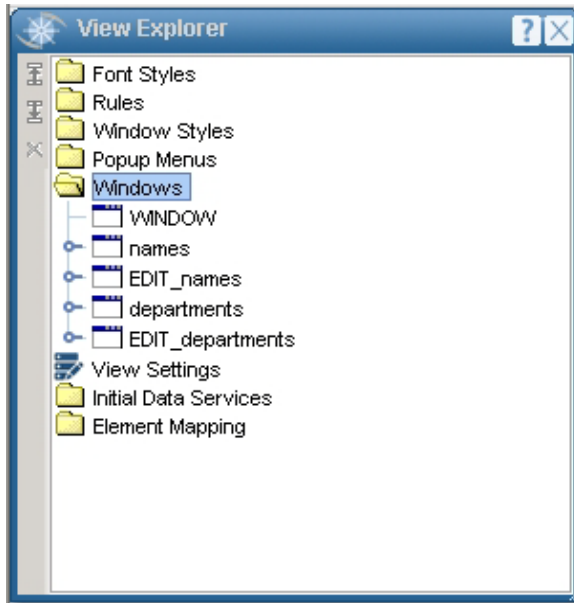
Now use the Designer to run the generated application and edit the automatically generated windows. On entering the Designer, remember to open the view first. You may also wish to run the Application Manager and examine the Service Functions, Datapools and/or Datakeys that have been generated.

13. Close the Prototyping Wizard and return to the AltioLive Studio.

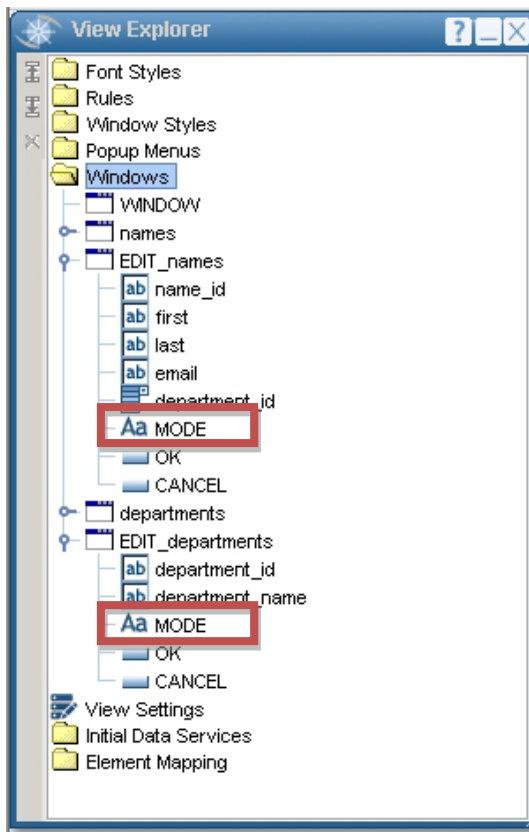
Now we will review the framework generated by the Wizard.

See the generated framework with the Designer

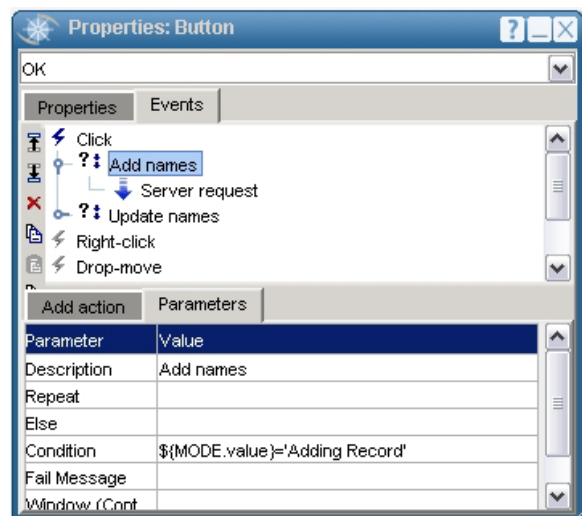
1. From the AltioLive Studio, select the **departments** project and click the **Designer** icon .
2. Look at the windows that have been created. Each element has a list window and an edit window:




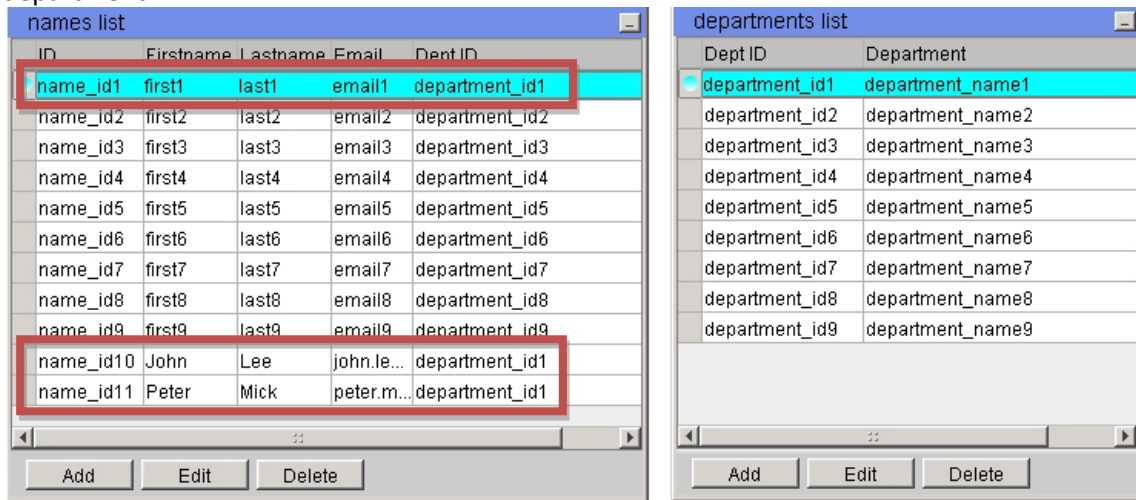
The **EDIT** windows have a hidden text control called **MODE**, used to determine if the window is in add or edit mode:



This then is used in the conditions on the **OK** buttons to call different Service Functions. The contents of these controls and the disabling of the **ID** fields are done in the **Click** actions of the list windows when the edit windows are invoked.



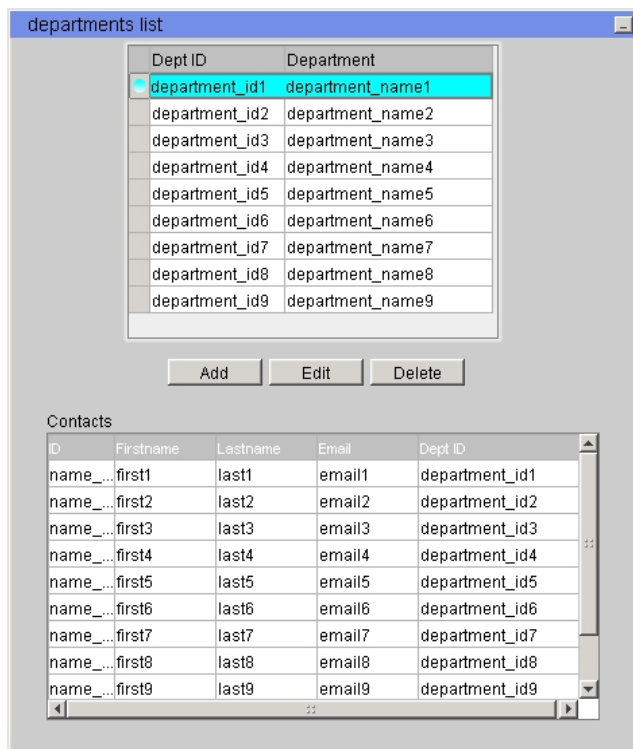
3. Test the Application by clicking on the **Run** icon .
4. Windows are presented, allowing names and department entries to be created, modified and deleted. Modify or add a couple of records so that more than one name is associated with a department:



5. If you have time, go to the Application Manager and look at the Service Functions, Datapools and Datakeys generated in the Application Manager.

Now we will create a window containing a Master-Detail list.

6. In the Designer, open the **departments** window.
7. Clone the window.
8. Double-click on the new window called **departments1** to display a preview of the window.
9. Adjust the size of the window and the position of the controls; then add a list control to display a Contacts detail list. The window should resemble the one shown here:



Remember that the easiest way to create the **Contacts** list is to display the **Datatypes** window and to drag and drop the attributes (**@name_id**, **@first**, **@last**, **@email** and **@department_id**) into the list.

10. Once the **Contacts** list created, save the view and close the Designer.

Creating Relationships with the Prototyping wizard

The session continues with a look at how we might work with the Prototyping Wizard. During this practical exercise, we will build Master-Detail lists that allow users to select a department on one list and view the names associated with the department on another.

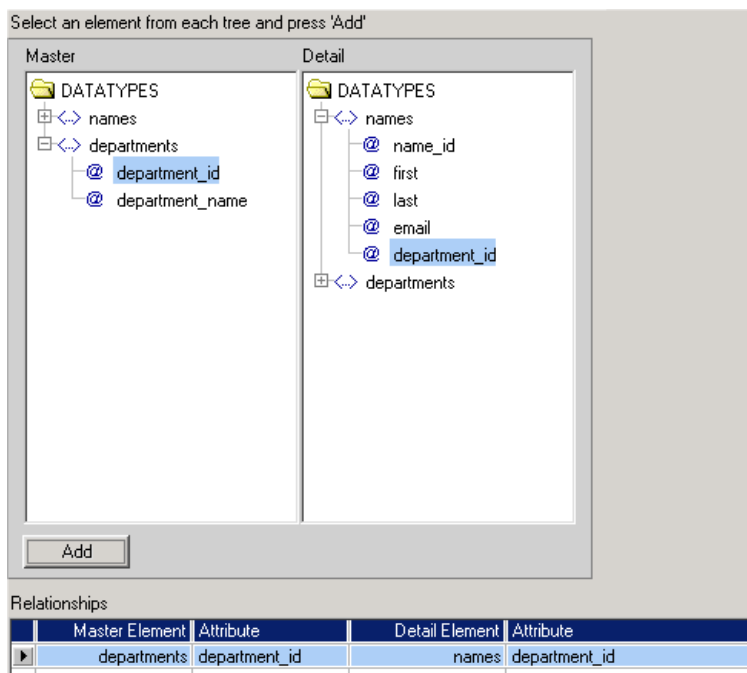
One of the features of the Developer environment is the ease with which we can ‘bind’ controls together. Our XML database allows this providing the relationship between elements is defined.

We will return to the Wizard to create the relationship and then return to the Designer to observe the change to the Master-Detail lists we have just defined.

1. From the AltioLive Studio, right-click on the **departments** project and select **Prototyping wizard** in the context menu.
2. Click on the **Define Relationships** tab.
3. In the **Master** column, expand the **departments** element and select the **department_id** attribute.
4. In the **Detail** column, expand the **names** element and select the **department_id** attribute.
5. Click on the **Add** button.


Here we define the relationship between the non-hierarchical XML data. Doing this before generating the framework will mean that Master-Detail lists can be created easily in the Designer.

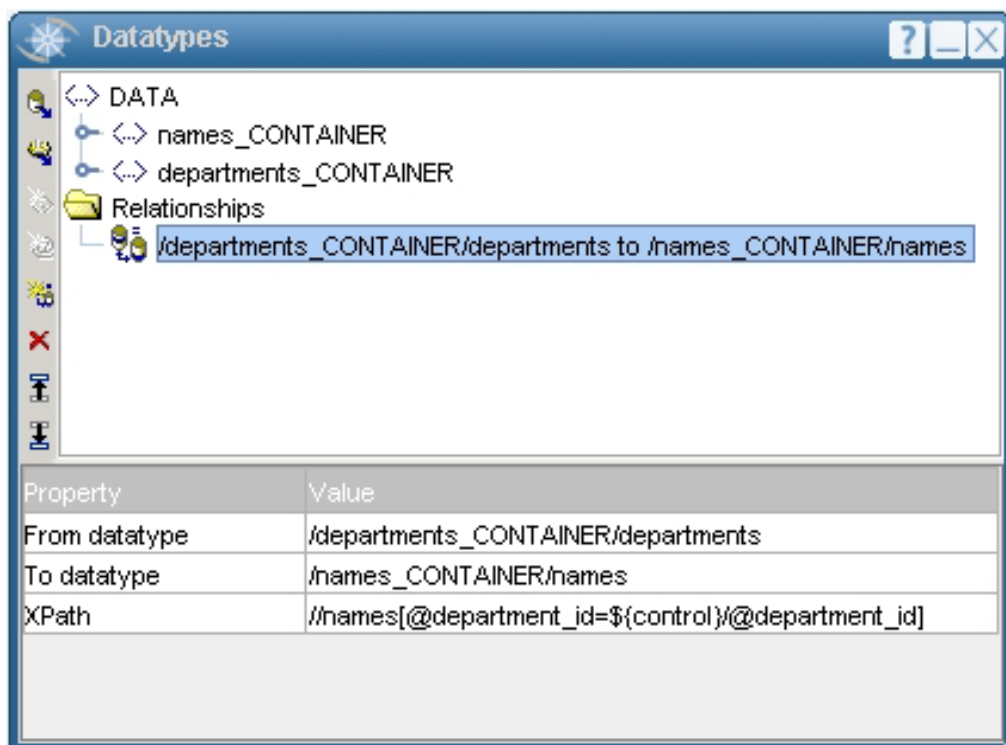
Here’s the definition finished:




The Master-Detail relationship is formed: One department may be linked to many names.

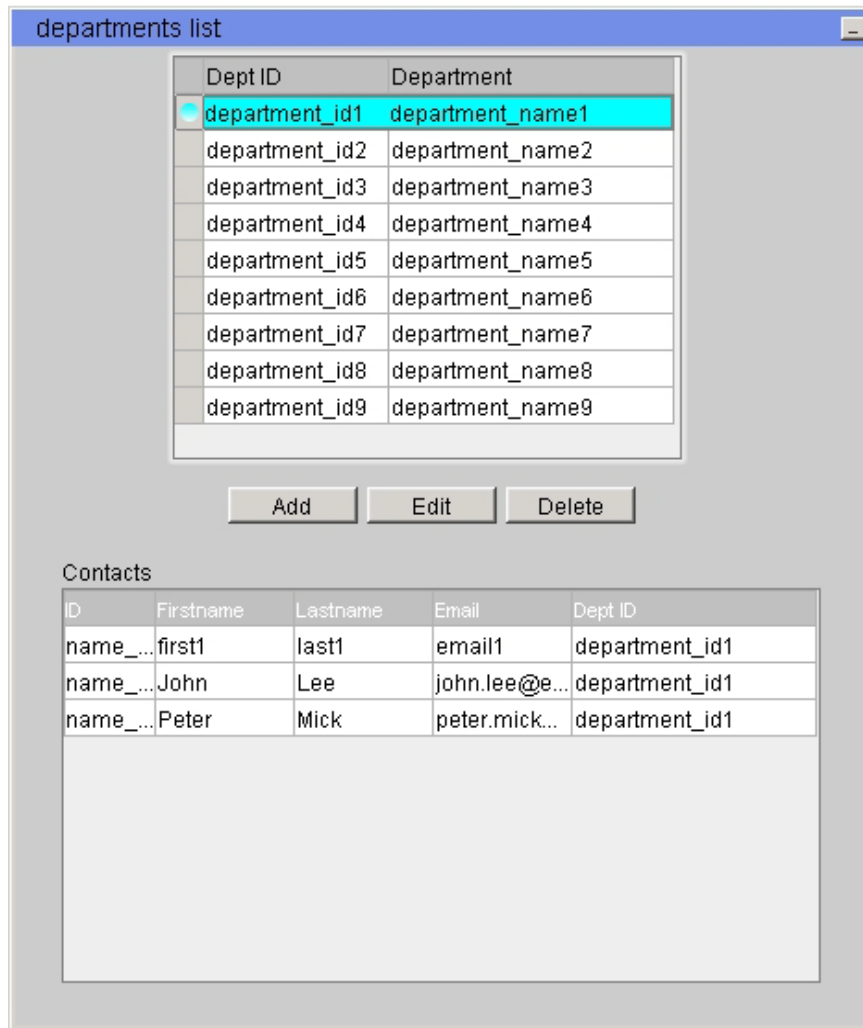
Please note: this process creates relationships that are used in the AltioLive Designer. Unlike relationships of relational databases they do not enforce referential integrity.

6. Select the **Build Prototype** tab and click the **Build** button.
7. **Confirm** the rebuild
The Wizard will recreate the framework. The Master-Detail window we created will be unaffected as it is not one of the default windows.
8. Exit the Prototyping Wizard and return to the Designer.
9. Open the **Datatypes** window by clicking on the **Show Datatypes** icon .
10. Look at the **Relationships** node. You will see that the relationship is now defined between the two datatypes. This is only necessary if the XML data structure is not hierarchical.



11. Assign the Master-Detail relationship by dragging, while holding the shift key, from the **departments** list to the **Contacts** list.
12. Test the application by clicking on the **Run** icon .

13. The **Contacts** list should display only the contacts matching the **department_id** selected in the list above:



Next Step: Debugging your Application

AltioLive Developer Training

Document Information

KEYWORDS: Prototyping Wizard, Framework, Relationships.

Integra SP – Altio

Telephone: +44 (0) 20 8528 1045

Internet: www.altio.com

Copyright © 2010 Integra SP

Copyright in this document is vested in Integra SP. The contents of the document (wholly or in part) must not be reproduced, distributed, used or disclosed without the prior written permission of Integra SP.

Integra recognizes the trademarks or registered trademarks of any third party product or company name referenced in this document at the time of its publication.