Dynamic Pages Manual

2.8

Table of Contents

1 Overview 3

2 Dynamic Page Selection 4

3 Formatting Dynamic Page - General 5

4 Formatting Dynamic Page – Text Region 6

4.4 Text File 7

4.5 Database Text 10

5 Formatting Dynamic Page – Image Region 20

5.4 Image List 21

5.5 Database Image 24

6 Formatter 32

7 Refresh / Update Interval 33

7.3 Refresh Now – Suite 34

7.4 Update Interval – Server 34

8 Rotations 34

9 Data Handling Options 44

10 Dynamic Page in a Sequence 49

Noventri Suite
Dynamic Pages Manual

1 Overview

1.1 The **Dynamic Page** feature enables a **Page** in a **Project** to ***Rotate*** through content that is pulled from an external **Data Source.** The number of ***Rotations*** is adjusted automatically to match the external **Data Source** volume, as it fluctuates.

1.2 Instead of having to create multiple **Pages,** that are nearly identical to each another. One **Dynamic Page** is created and formatted to accommodate the **Rotation** of content from an external **Data Source**.

1.3 Any **Page** or number of **Pages** within a **Project** can be set up as a **Dynamic Page** to allow flexibility in the **Project** design.

1.4 Once a **Page** is chosen as **Dynamic Page** the **Text / Image Regions** on that **Page** are **Dynamic Regions** that can point to and **Rotate** through content from a fluctuating **Data Source**.

1.5 A **Dynamic Page** can contain multiple **Dynamic Text / Image Regions**. These **Dynamic Text / Image Regions** can also easily be configured to be used as normal **Text / Image Regions** that do not **Rotate** their content.

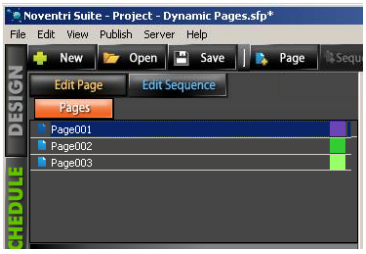
1.6 **Dynamic Regions** can be configured to access multiple content from numerous types of **Data Sources**.

1.7 The **Dynamic Regions** have options that allow the data from the external **Data Source** to be *selectively* accessed.

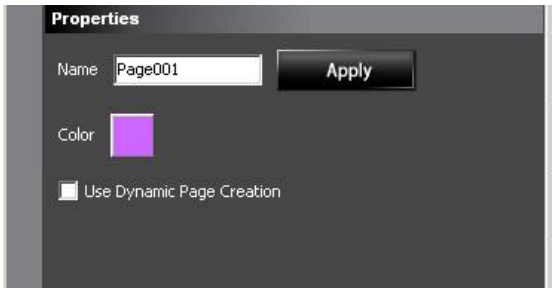
2 Dynamic Page Selection

2.1 A new **Project** can be created or an existing **Project** can be used.
(see the **Design** manual)

2.2 Once a **Project** is open in the **Design** tab of the **Noventri Suite**, choose the **Page** that will be the **Dynamic Page**.



2.3 When the **Page** is selected, it's properties will be shown in the **Properties Panel** (bottom left).



Note: If a **Region** happens to be selected, the ***Region*** **Properties Panel** will be displayed instead of the ***Page*** **Properties Panel**. To switch to the ***Page*** **Properties Panel**, click the mouse outside of any **Region** boundaries.

In some cases when the **Page** is completely covered with **Region/s** it will be necessary to temporarily select another **Page** in the **Project** and then go back to the desired page to display it's ***Page*** **Properties Panel**.

2.4 Within the **Properties Panel** there is the **Use Dynamic Page Creation** option. Selecting this check box will activate this **Page** as a **Dynamic Page**.

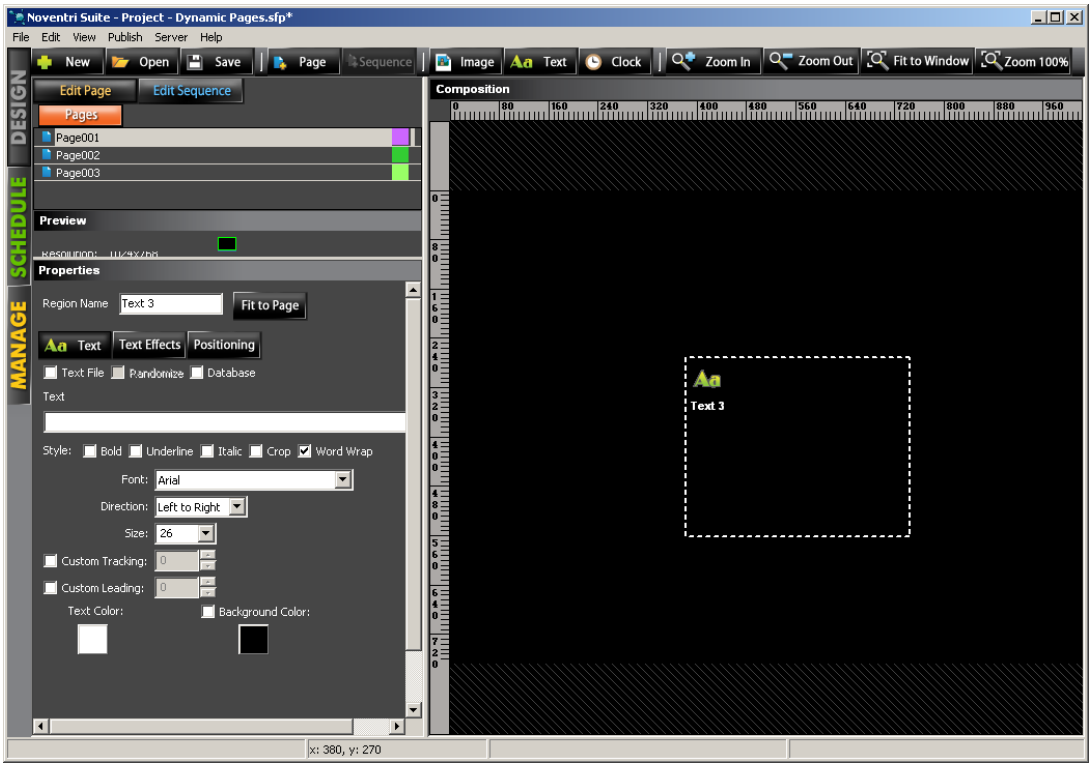
***Note:** The **Use Dynamic Page Creation** check box needs to be selected to ensure that all **Dynamic Region Properties** options are visible in the **Region Properties** panel.

3 Formatting Dynamic Page - General

3.1 Once a **Page** is a **Dynamic Page**, it's existing and newly created **Text/Image Regions** can be used as **Dynamic Regions**.

3.2 Any **Text/ Image Region** that will have it's content **Dynamically** displayed (**Rotated**) needs to have it's **Properties** configured to access an external **Data Source**.

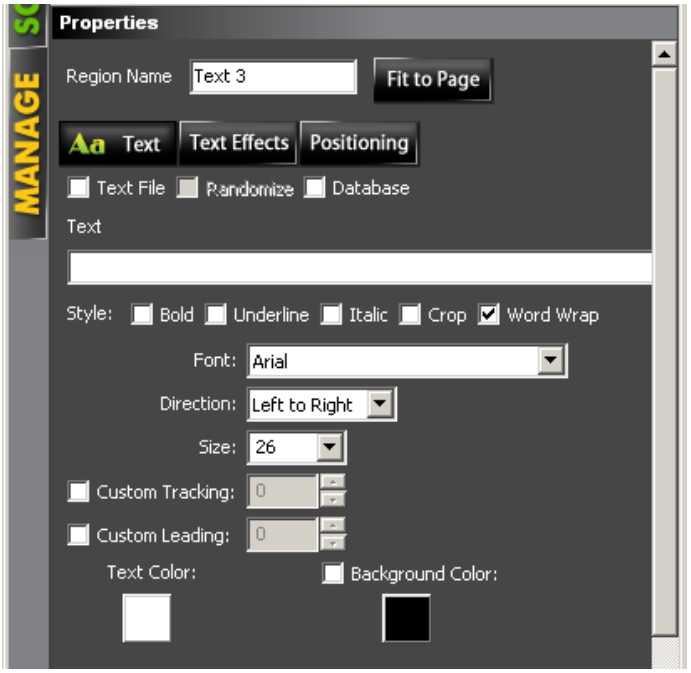
3.3 When the **Region** is selected, it's properties will be shown in the **Properties Panel** (bottom left).



4 Formatting Dynamic Page – Text Region

4.1 For a **Text Region** to be utilized as a **Dynamic Region** (rotating content) within the **Dynamic Page**, it needs to be pointed to an external **Data Source** that contains multiple **Text** entries.

4.2 The **Text Region's Properties Panel** would be selected as it normally would be when configuring a **Text Region**.



4.3 **Text Regions** can be pointed to an external **Data Source** in two different ways, through a **Text File** or through the **Database** option.

4.4 Text File

4.4.1 This option allows a **Text File** to be used as the **Data Source** for the content of **Dynamic Region/s** on a **Dynamic Page**.

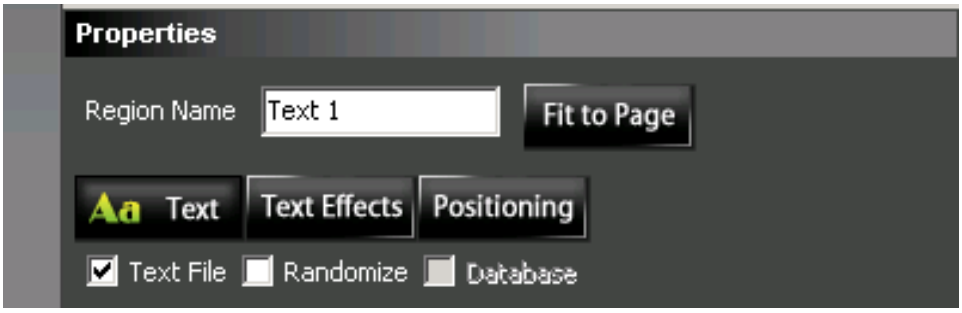
Note: A **Text File** is a *list* of **Text** entries.
(covered here)

where as....

A **Data Text** file is a text file created with *delimiters.*
(see the **Database** section of this manual)

4.4.2 Each line of text in the **Text File** will cause the **Dynamic Page** to be displayed again - populated with that text. This will continue until the last line of **Text** has been displayed.

4.4.3 Select the **Text File** option by selecting the **Text File** check box.

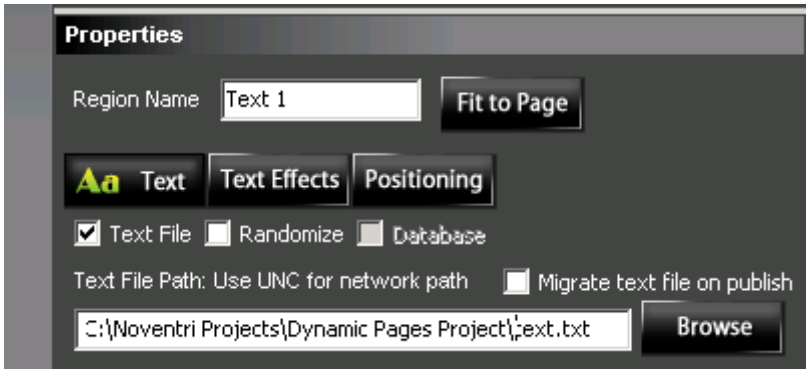


4.4.4 Set up a **Text File** (.txt) that contains a list of the **Text** entries to be displayed.
(this can be done in any text program such as Notepad)

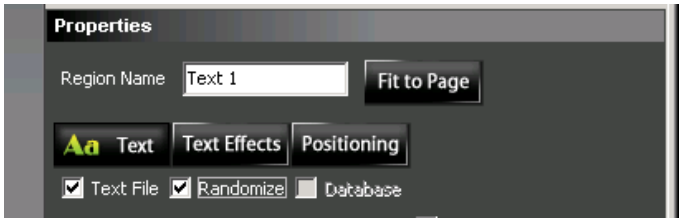
Example File:
Text File Line 1
Text File Line 2
Text File Line 3

Metacharacters can also be used within the **Text File**.
(see the **Data Handling Options** section of this manual)

4.4.5 In the **Text Properties** panel fill in the **Text File Path** or use the **Browse** button to select it.

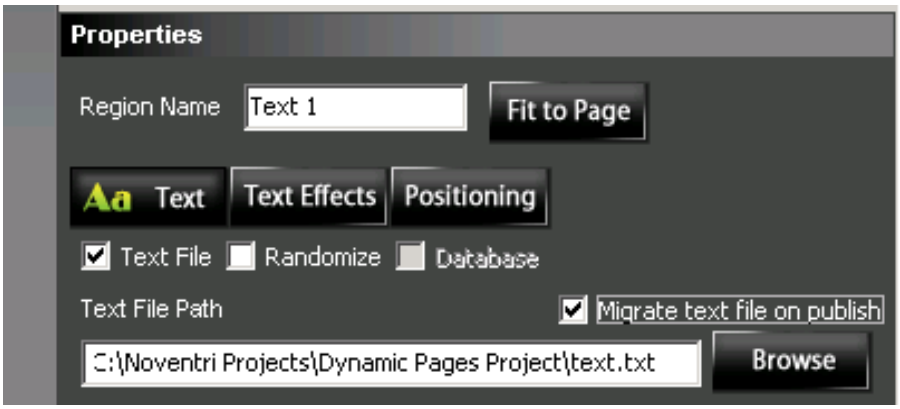


4.4.6 **Randomize** – Normally **Dynamic Page** creation for a **Text File** will cause **Rotations** that will display the **Text** lines in the order they are listed in the **Text File**. To allow this process to be random, select the **Randomize** check box.



Note: When **Randomize** is selected the number of **Rotations** will still match the number of **Text** lines in the **Text File**, however some **Text** lines may display more than once, and other **Text** lines may never be displayed. (truly random)

4.4.7 Migrate Text File on Publish – Lets a choice be made between coping the **Text File** to the **Server** when the **Project** is **Published**, or if allowing the **Server** to access the **Text File** at it's present location.



Migrate Text File on Publish *Checked*:

Selecting the **Migrate Text File on Publish** option will place a copy of the image on the **Server**. This is typically the **Noventri Suite Projects** directory located on the PC where the **Server** is installed.

Advantages:

- No need to worry about the original path remaining available.
- **Text File** is less likely to be accidentally modified, deleted, or moved.

Migrate Text File on Publish *Unchecked*:

The **Text File** will be accessed by the **Server** at it's current location. The **Text File** will not be copied to the **Server** with the **Project** when it is published. The **Text File** location must remain available or the text will not be displayed.

Advantages:

- Changing the text in the **Text File** can be done at its source (as long as the file name remains the same) without having to republish the **Project**.

4.4.8 When setting up **Data** to be displayed in a **Dynamic Region**, there are a multitude of creative ways the **Data** can be handled and displayed.
(see the **Data Handling Options** section of this manual)

4.4.9 The **Rotations** for this **Region** will display a different row of data from the **Data Source** (as set up with **Row offset** and **Row Increment**). The number of **Rotations** will depend on how many rows of data are in the **Data Source** as well as how many other **Dynamic Regions** are on the **Dynamic Page**.
(see the **Rotations** section of this manual)

4.4.10 When developing your **Dynamic Region** the **Data** from the external **Data Source** can be inserted and viewed in the **Suite Design** window.
(see the **Refresh Now** / **Update Interval** section of this manual)

4.4.11 The content of each **Rotation** of a **Dynamic Page** can also be viewed in context of how it will be displayed with other **Pages**, by adding the **Dynamic Page** to a **Sequence**.
(see the **Adding a Dynamic Page to a Sequence** section of this manual)

4.5 Database Text

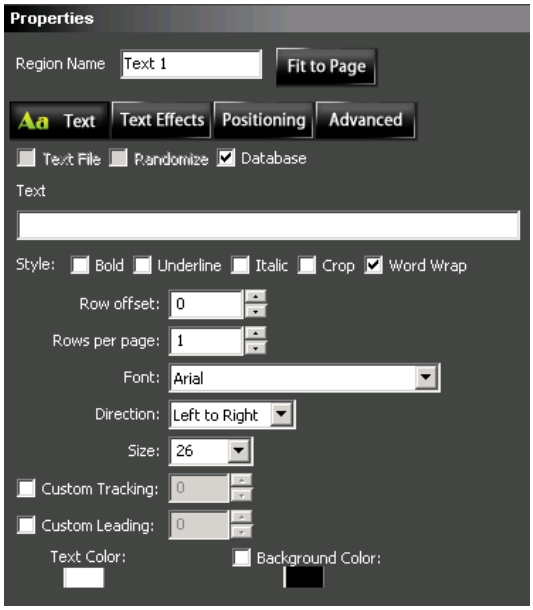
4.5.1 The **Database** option allows the use of many different types of **Data Sources**. All of which are stored in a **Database** style. This can be a simple **Text File** that has been created with delimiters, a **Spreadsheet**, an **XLM(RSS)** source, or a true **Database**.
(see the **Noventri Suite - Database Manual**)

Note: Performance and reliability is best when using a true **Database** for the **Data source** as opposed to a Excel **Spreadsheet**.

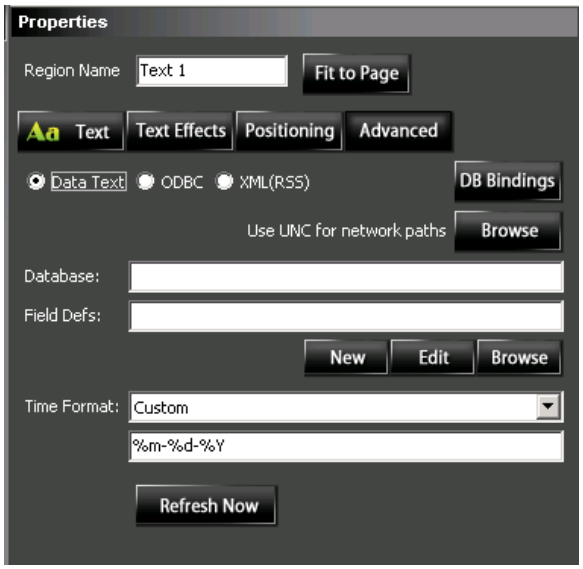
4.5.2 The **Dynamic Region/s** on the **Dynamic Page** connect to the **Data Source** and pull the data to be displayed.

4.5.3 Each row of text in the **Data Source** (as selected by the configuration) will cause a **Rotation** of the **Dynamic Page** populated with that data. This will continue until the last row of **Text** has been displayed.

4.5.4 A **Database Text Region** is created by selecting the **Database** check box.



4.5.5 Once the **Database** option is selected a new **Advanced** tab will appear in the **Properties** panel. Selecting this **Advanced** tab will display the **Advanced** options.

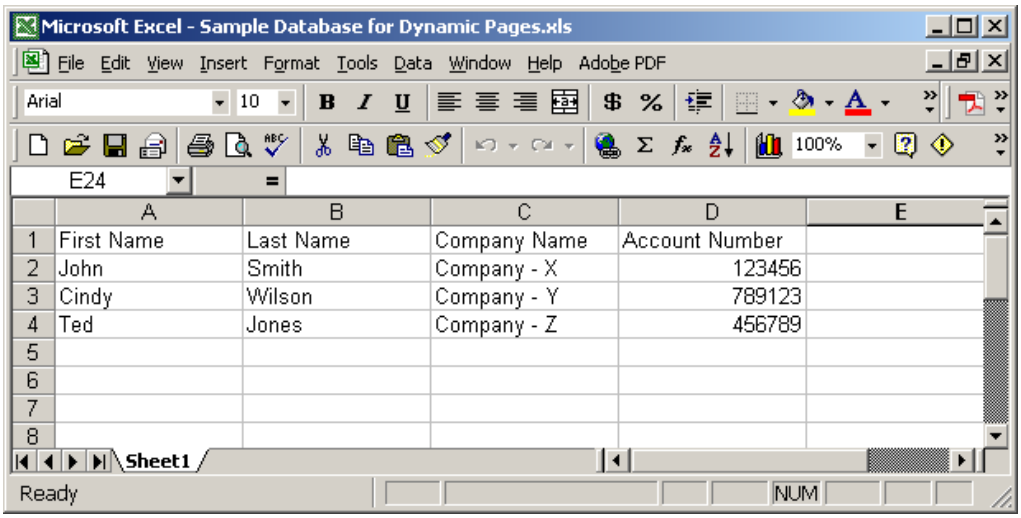


Example using Excel Spreadsheet

4.5.6 For the purposes of this **Dynamic Pages Manual**, a simple Excel **Spreadsheet** will be used to demonstrate the **Database** use of **Dynamic Pages**, however **Dynamic Pages** is fully compatible with all the **Database** capabilities. Many **Database** options that are available with **Dynamic Pages** are not covered in this manual.
(see the **Noventri Suite Database Manual**)

Note: When using a Excel **Spreadsheet** be sure not to use the first row for data to be displayed. The first row is always assumed to be column headers and is therefore not accessible.

Sample Excel Spreadsheet

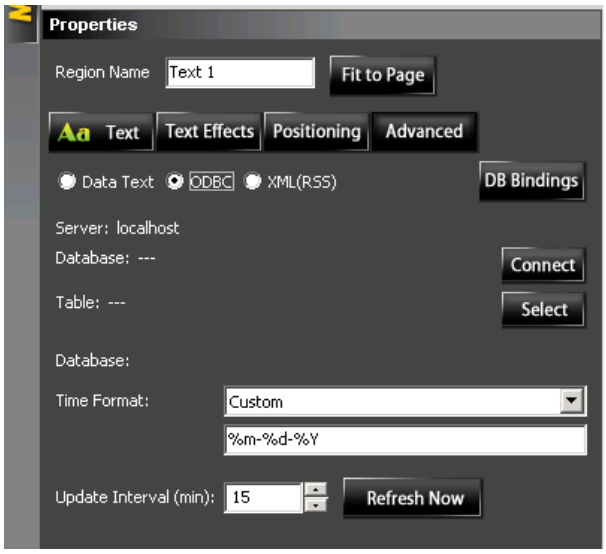


	A	B	C	D	E
1	First Name	Last Name	Company Name	Account Number	
2	John	Smith	Company - X	123456	
3	Cindy	Wilson	Company - Y	789123	
4	Ted	Jones	Company - Z	456789	
5					
6					
7					
8					

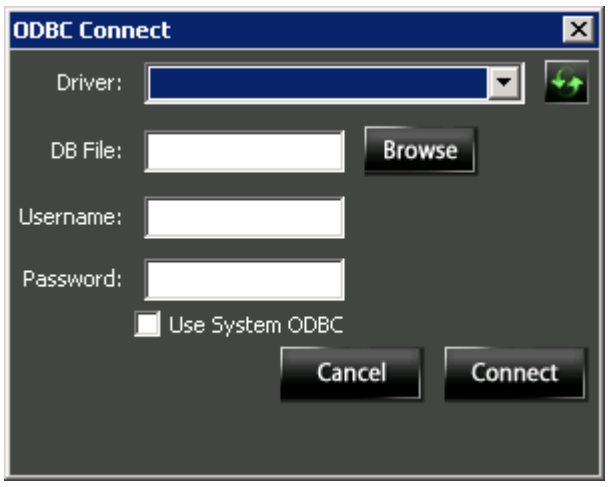
4.5.7 Under the **Advanced** tab select the **ODBC** (Open Database Connectivity) option.

Example using Excel Spreadsheet

4.5.8 Once the **ODBC** option is selected, the **Advanced** tab menu options will change to reflect it. To connect to the **Spreadsheet** select the **Connect** button.

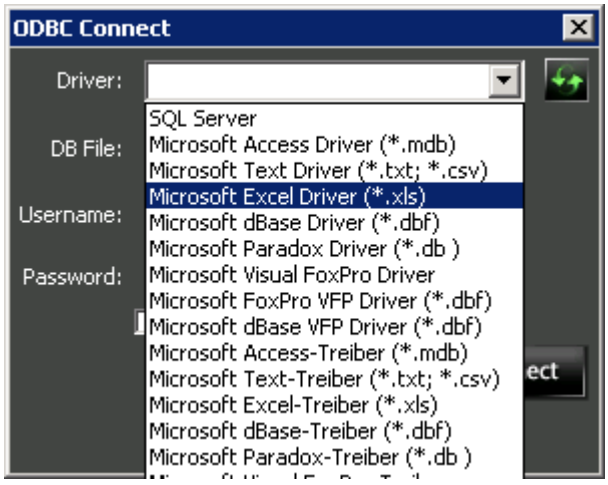


4.5.9 The **ODBC Connect** window will then open.



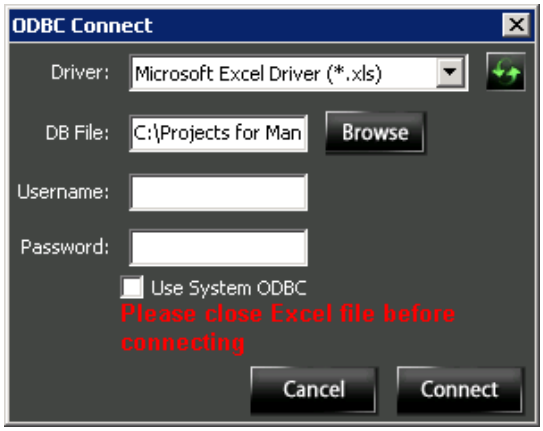
Example using Excel Spreadsheet

4.5.9.1 **Driver** – Select the desired type of **Database** to be connect to using the **Driver** drop down menu. For this example we are using a **Spreadsheet** created with Excel so the **Driver** **"Microsoft Excel Driver (*.xls)"** needs to be selected.



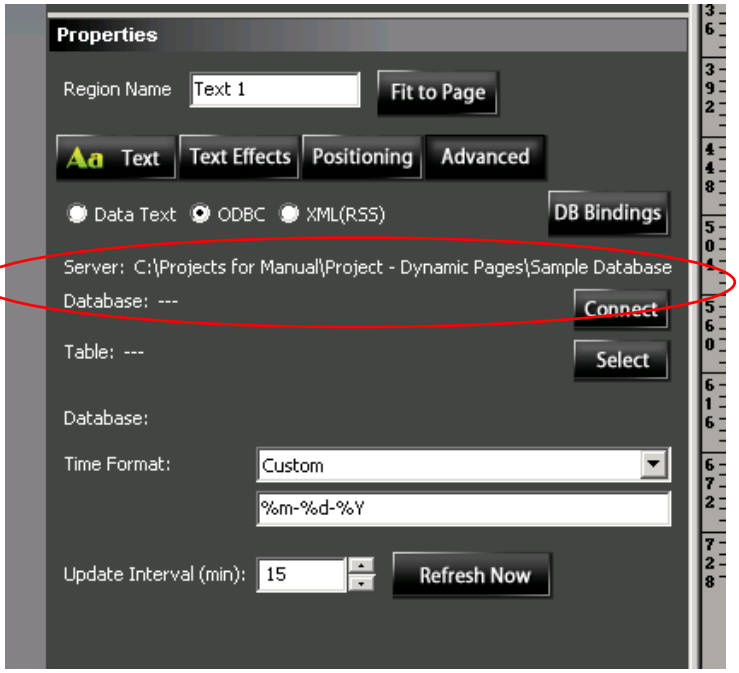
4.5.9.2 **DB File** – Browse to the **Database** file and select **Open**.

Note: Some **Database** files, including Excel files need to be closed before they can be connected to.



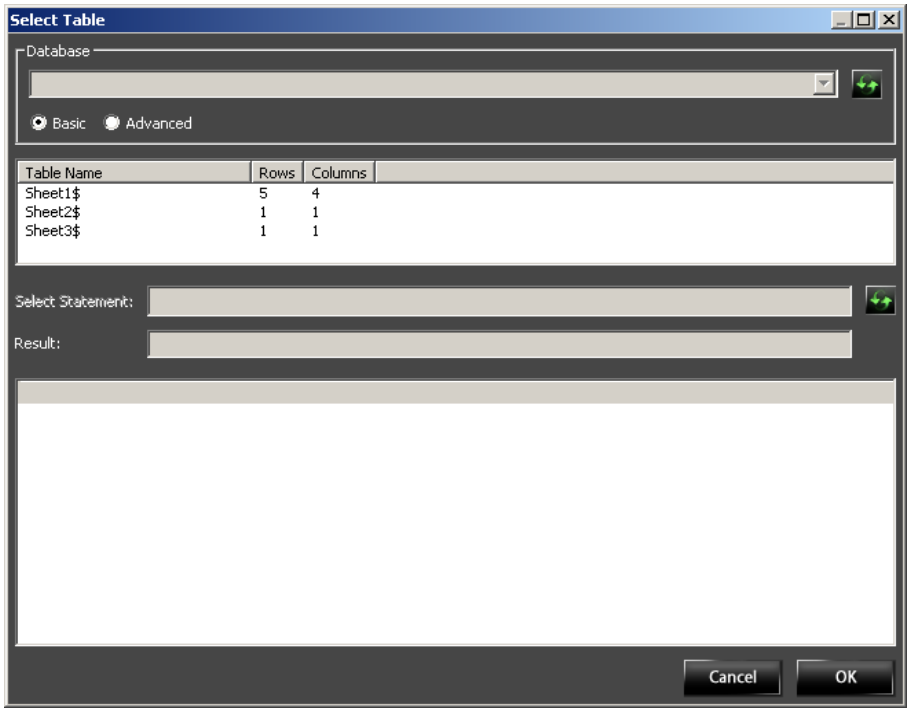
Example using Excel Spreadsheet

4.5.9.3 **Connect** – Select the **Connect** button. There will now be a connection to the **Database** and the connection can be verified by seeing the path to the file shown as the **Server**, under the **Advanced** tab.



Example using Excel Spreadsheet

4.5.10 Select – Select the **Select** button. The **Select Table** window will open.



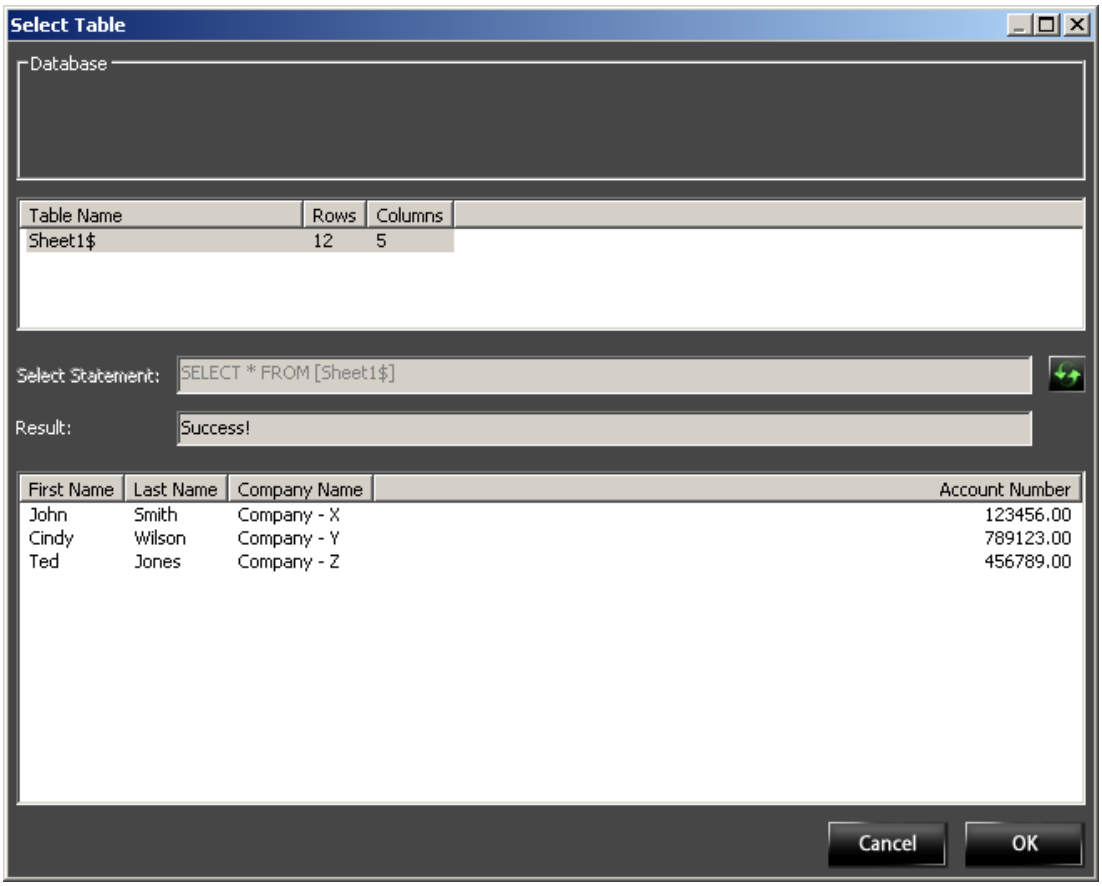
4.5.10.1 Select the **Table Name** that contains the content to be use for this **Text Region.**

Note: For Excel files the **Table Name** represents the Sheets of the Excel document.

4.5.10.2 To populate the **Database** content into the viewing area, select the **Refresh** button located to the right of the **Select Statement** box.

Example using Excel Spreadsheet

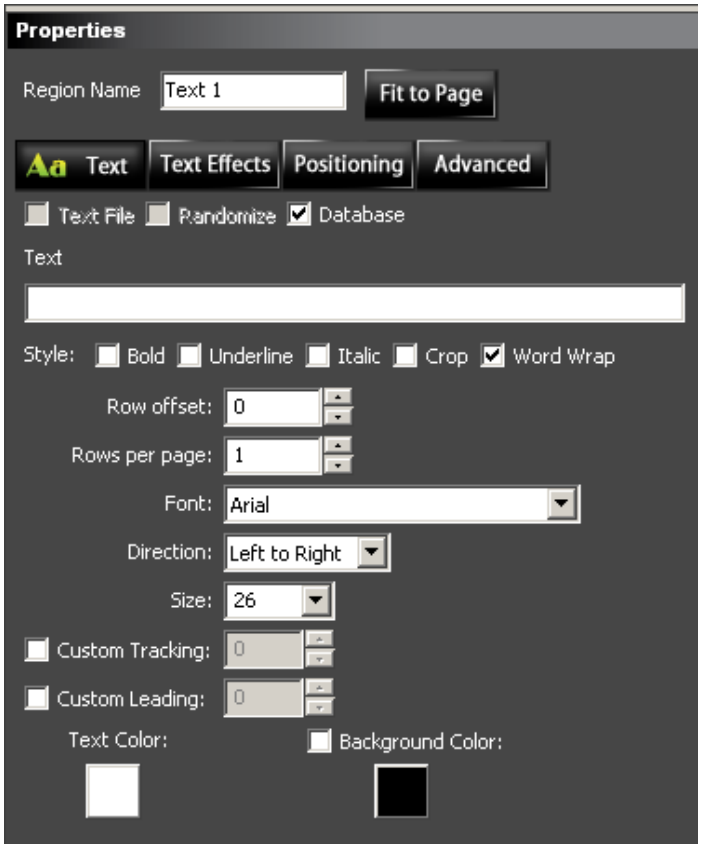
4.5.10.3 The content of the **Table** will now be visible in the viewing area.



4.5.10.4 The **Table** is now linked to the **Region**.
Select **OK** to close the **Select Table** window.

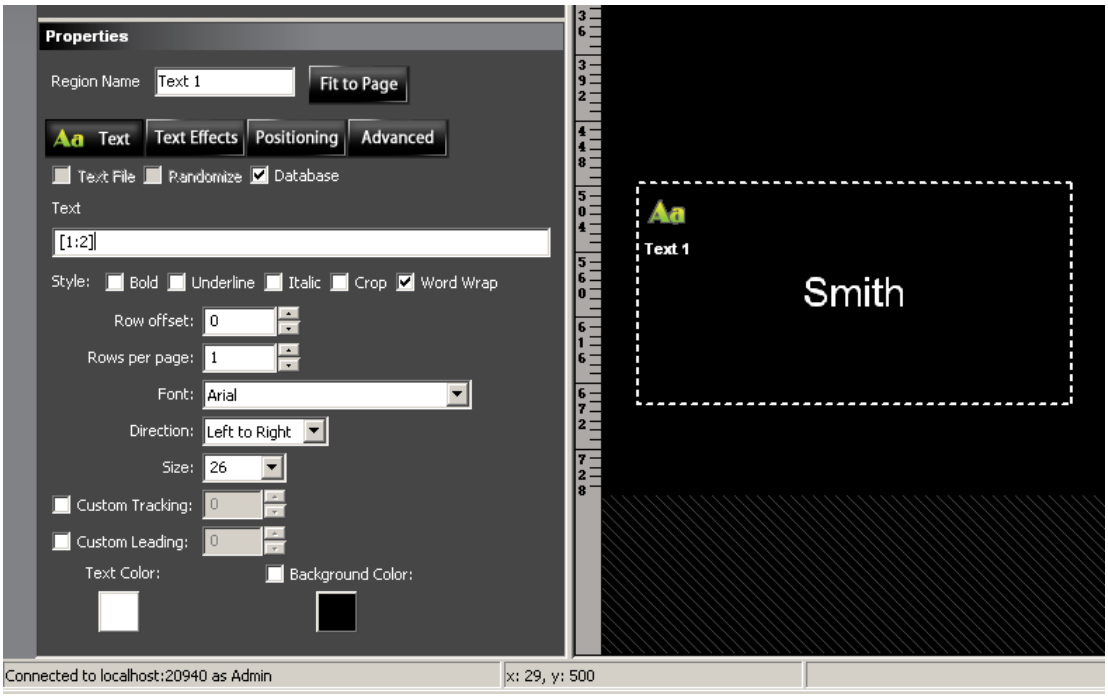
4.5.11 Now that the **Table** is linked to the **Dynamic Region**, the **Dynamic Region** can be formatted to display specific data from the **Table**.

4.5.12 Return to the **Text** tab in the **Properties** panel.



4.5.13 Text box **-** Now that the **Text Region** is properly connected to an external **Data Source**, the **Text** box, can be used to point at the **Database** content using a **Formatter**.
(see the **Formatter** section of this manual)

Note: The **Text** box when used for **Dynamic Pages** can (in addition to the **Formatter**) contain typed text, **Metacharacters**, and special **Dynamic Page** variables.
(see the **Data Handling Options** section of this manual)



4.5.14 When setting up **Data** to be displayed in a **Dynamic Region**, there are a multitude of creative ways the **Data** can be handled and displayed.
(see the **Data Handling Options** section of this manual)

4.5.15 The **Rotations** for this **Region** will display a different row of data from the **Data Source** (as set up with **Row offset** and **Row Increment**). The number of **Rotations** will depend on how many rows of data are in the **Data Source** as well as how many other **Dynamic Regions** are on the **Dynamic Page**.
(see the **Rotations** section of this manual)

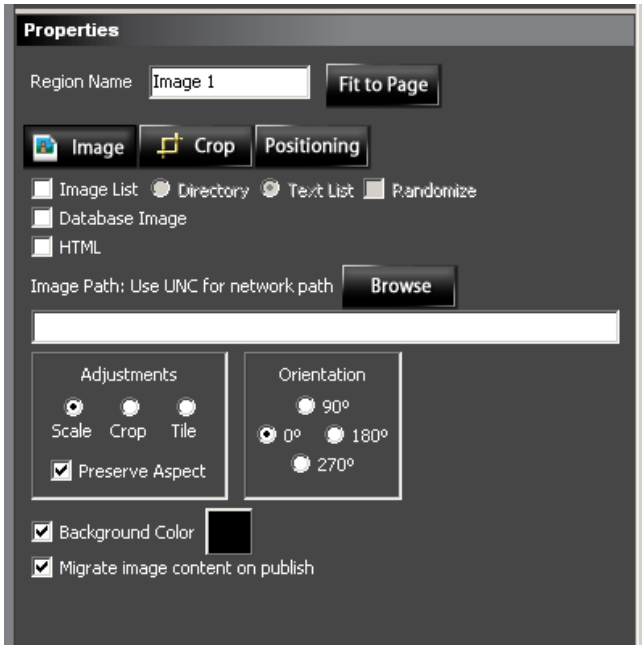
4.5.16 When developing your **Dynamic Region** the **Data** from the external **Data Source** can be inserted and viewed in the **Suite Design** window.
(see the **Refresh Now** / **Update Interval** section of this manual)

4.5.17 The content of each **Rotation** of a **Dynamic Page** can also be viewed in context of how it will be displayed with other **Pages**, by adding the **Dynamic Page** to a **Sequence**.
(see the **Adding a Dynamic Page to a Sequence** section of this manual)

5 Formatting Dynamic Page – Image Region

5.1 For an **Image Region** to be utilized as a **Dynamic Region** (rotating content) within a **Dynamic Page**, it needs to be pointed to an external **Data Source** that contains paths to multiple **Image** entries.

5.2 The **Image Region's Properties Panel** would be selected as it normally would be when configuring an **Image Region**.

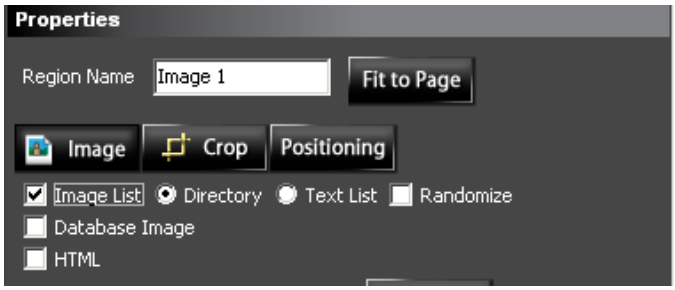


5.3 **Image Regions** can be pointed to a **Data Source** in two different ways, through the **Image List** option, or the **Database Image** option.

5.4 Image List

5.4.1 This option allows the use of a list of **Image** paths. The paths will direct the **Dynamic Region** to the **Image** files one at a time. Each **Rotation** will be populated with an **Image**. This will continue until the last **Image** has been displayed.

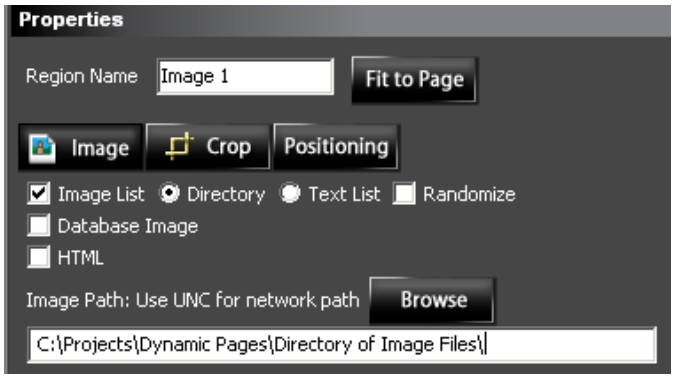
5.4.2 Select this option by selecting the **Image List** check box.



5.4.3 An **Image List** can be done in two different ways, through the **Directory** option or through the **Text List** option.

5.4.4 Directory - This option allows a file **Directory** to be used as the **Data Source**. The **Directory** would contain the actual **Image** files.

5.4.4.1 In the **Image Properties** panel select the **Directory** option by selecting the **Directory** radio button.



5.4.4.2 In the **Text Properties** panel fill in the **Image Path** box or use the **Browse** button to select the **Directory** that contains the **Image** files.

Example Path:

c:\Projects\Dynamic Pages\Directory of Image Files\

5.4.5 Text List - This option allows a .txt file to be used as the **Data Source** (list of **Image** paths).

Note: A **Text List** is a *list* of **Image** path entries. (covered here)

where as....

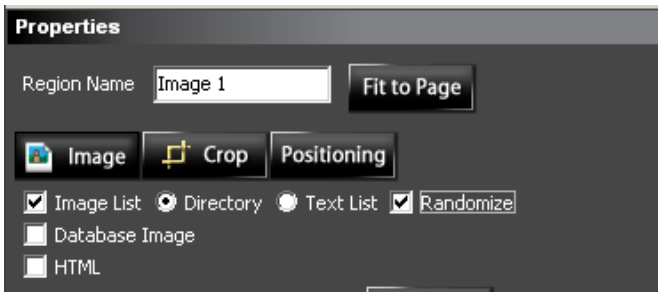
A **Data Text** file is a text file created with *delimiters.* (see the **Database** section of this manual)

5.4.5.1 Set up a **Text List** file (.txt) that contains a list of the **Image** paths.
(this can be done in any text program such as Notepad)

Example File:

c:\Projects\Dynamic Pages\Directory of Image Files\Image 1.jpg
c:\Projects\Dynamic Pages\Directory of Image Files\Image 2.jpg
c:\Projects\Dynamic Pages\Directory of Image Files\Image 3.jpg
c:\Projects\Dynamic Pages\Directory of Image Files\Image 4.jpg

5.4.6 **Randomize** – Normally **Dynamic Page** creation for **Images** will **Rotate** each **Image** in the order they are listed in the **Image List** (**Directory/Text List**). To allow this process to be random, select the **Randomize** check box.



Note: When **Randomize** is selected the number of **Rotations** will still match the number of **Images** in the list, however some **Images** may be displayed more than once, and other **Images** may never be displayed. (truly random)

5.4.7 When setting up **Data** to be displayed in a **Dynamic Region**, there are a multitude of creative ways the **Data** can be handled and displayed.
(see the **Data Handling Options** section of this manual)

5.4.8 The **Rotations** for this **Region** will display a different row of data from the **Data Source** (as set up with **Row offset** and **Row Increment**). The number of **Rotations** will depend on how many rows of data are in the **Data Source** as well as how many other **Dynamic Regions** are on the **Dynamic Page**.
(see the **Rotations** section of this manual)

5.4.9 When developing your **Dynamic Region** the **Data** from the external **Data Source** can be inserted and viewed in the **Suite Design** window.
(see the **Refresh Now** / **Update Interval** section of this manual)

5.4.10 The content of each **Rotation** of a **Dynamic Page** can also be viewed in context of how it will be displayed with other **Pages**, by adding the **Dynamic Page** to a **Sequence**.
(see the **Adding a Dynamic Page to a Sequence** section of this manual)

5.5 Database Image

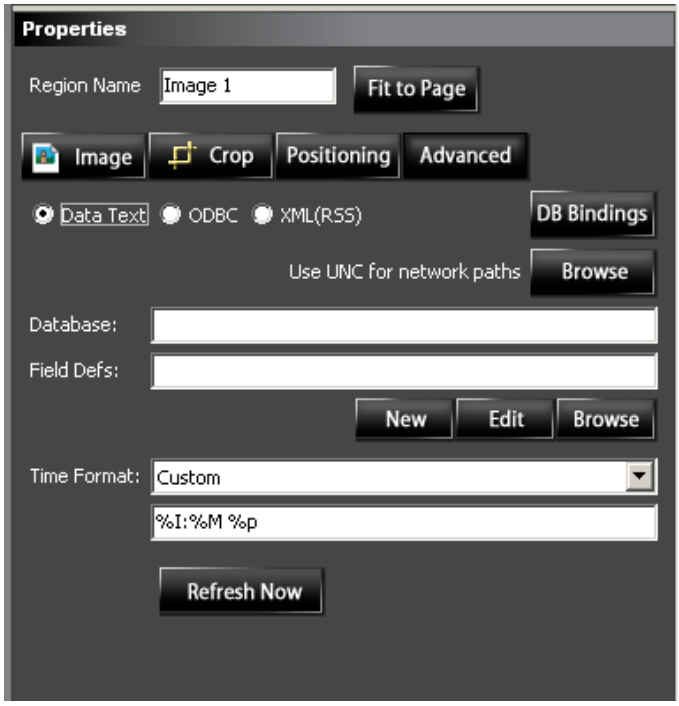
5.5.1 The **Database** option allows the use of many different types of **Data Sources**. All of which are stored in a **Database** style. This can be a simple **Text File** that has been created with delimiters, a **Spreadsheet**, an **XLM(RSS)** source, or a true **Database**.
(see the **Noventri Suite - Database Manual**)

Note: Performance and reliability is best when using a true **Database** for the **Data source** as opposed to an Excel **Spreadsheet**.

5.5.2 Each row of Image paths in the **Data Source** (as selected by the configuration) will cause a **Rotation** of the **Dynamic Page** diplaying that **Image**. This will continue until the **Image** has been displayed.

5.5.3 A **Database Image Region** is created by selecting the **Database Image** check box.

5.5.4 Once the **Database** option has been selected a new **Advanced** tab will appear in the **Properties** panel. Selecting this **Advanced** tab will display the **Advanced** options.

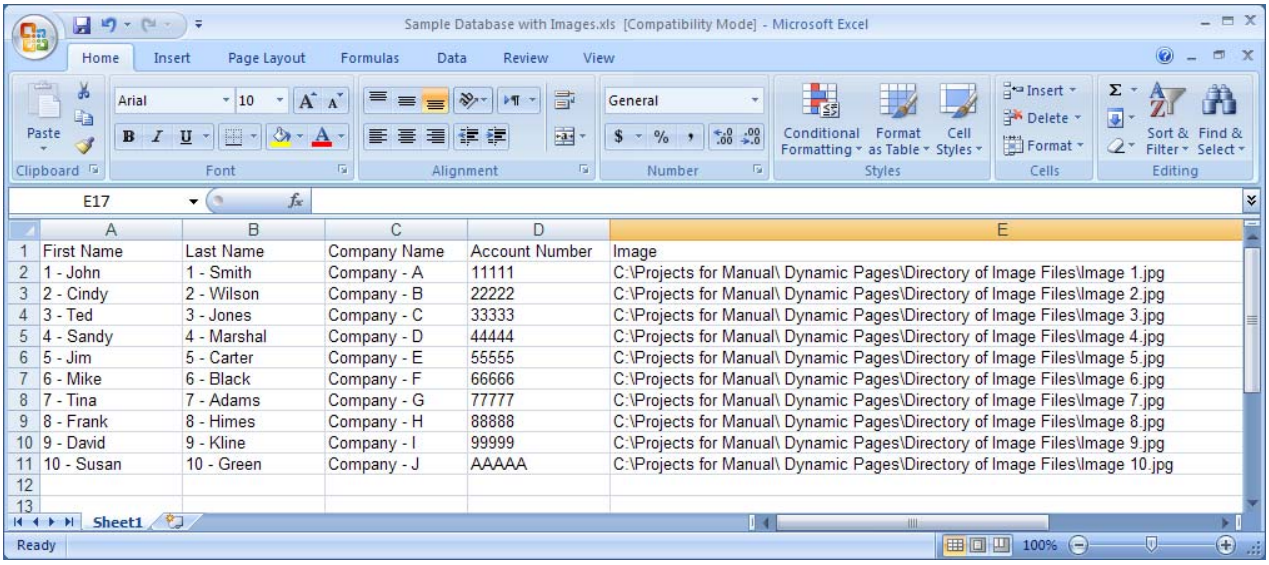


Example using Excel Spreadsheet

5.5.5 For the purposes of this **Dynamic Pages Manual**, a simple Excel **Spreadsheet** will be used to demonstrate the **Database** use of **Dynamic Pages**, however **Dynamic Pages** is fully compatible with all the **Database** capabilities. Many **Database** options that are available with **Dynamic Pages** are not covered in this manual.
(see the **Noventri Suite Database Manual**)

Note: When using an Excel **Spreadsheet** be sure not to use the first row for **Images** to be displayed. The first row is always assumed to be column headers and is therefore not accessible.

Sample Excel Spreadsheet with Images

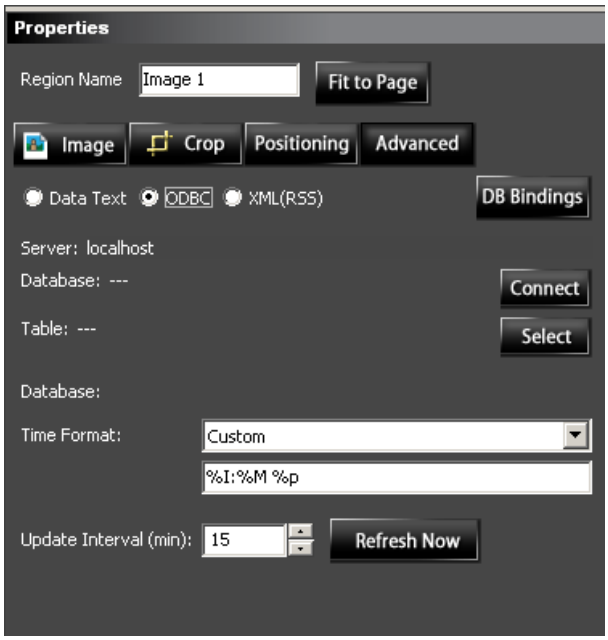


	A	B	C	D	E
1	First Name	Last Name	Company Name	Account Number	Image
2	1 - John	1 - Smith	Company - A	11111	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 1.jpg
3	2 - Cindy	2 - Wilson	Company - B	22222	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 2.jpg
4	3 - Ted	3 - Jones	Company - C	33333	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 3.jpg
5	4 - Sandy	4 - Marshal	Company - D	44444	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 4.jpg
6	5 - Jim	5 - Carter	Company - E	55555	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 5.jpg
7	6 - Mike	6 - Black	Company - F	66666	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 6.jpg
8	7 - Tina	7 - Adams	Company - G	77777	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 7.jpg
9	8 - Frank	8 - Himes	Company - H	88888	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 8.jpg
10	9 - David	9 - Kline	Company - I	99999	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 9.jpg
11	10 - Susan	10 - Green	Company - J	AAAAA	C:\Projects for Manual\ Dynamic Pages\Directory of Image Files\Image 10.jpg
12					
13					

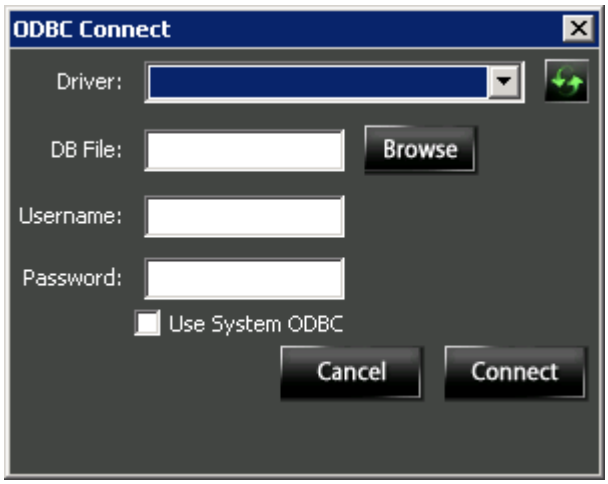
Example using Excel Spreadsheet

5.5.6 Under the **Advanced** tab select the **ODBC** (Open Database Connectivity) option.

5.5.7 Once the **ODBC** option is selected, the **Advanced** tab menu options will change to reflect it. To connect to the **Spreadsheet** select the **Connect** button.

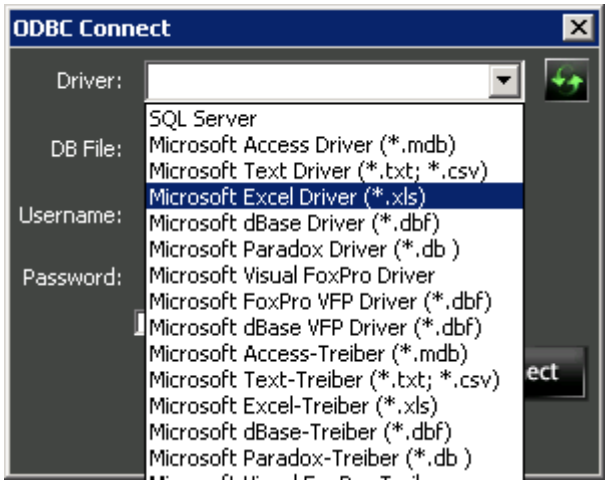


5.5.8 The **ODBC Connect** window will then open.



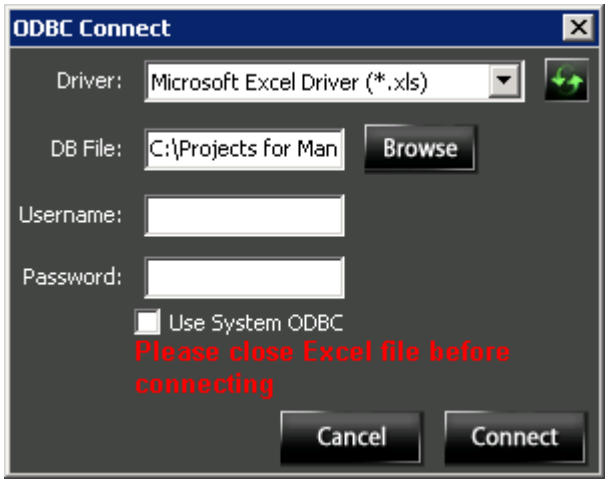
Example using Excel Spreadsheet

5.5.8.1 **Driver** – Select the **Driver** drop down menu and select the type of **Database** that will be connected to. For this example we are using a **Spreadsheet** created with Excel so the **Driver "Microsoft Excel Driver (*.xls)"** needs to be selected.



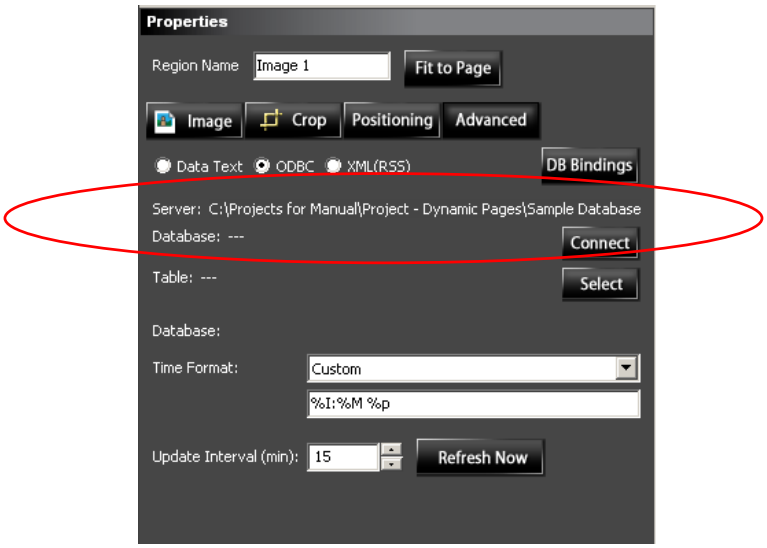
5.5.8.2 **DB File** – Browse to the **Database** file and select **Open**.

Note: Some **Database** files, including Excel files need to be closed before they can be connected to.

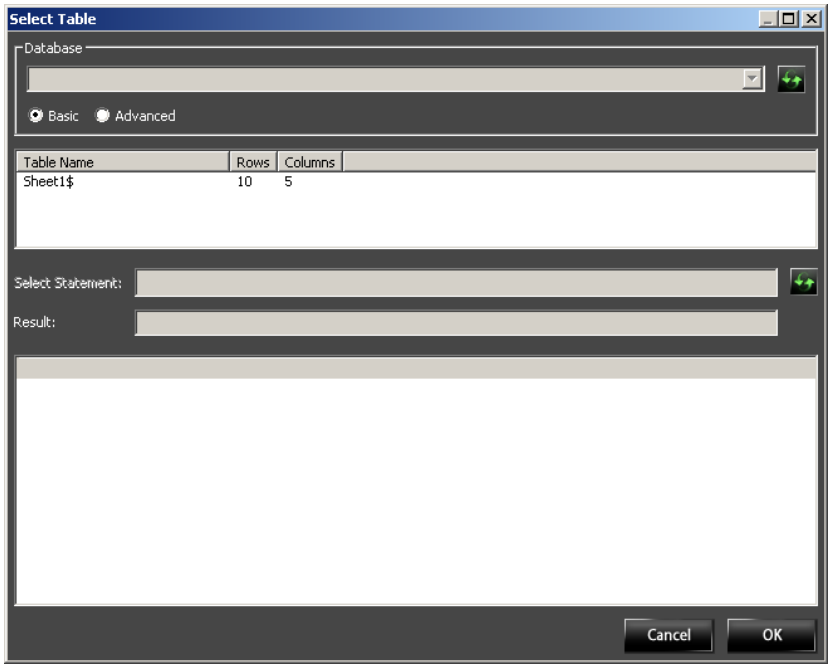


Example using Excel Spreadsheet

5.5.8.3 **Connect** – Select the **Connect** button. There will now be a connection to the **Database** and to verify that the connection has been made, the path to the file is shown as the **Server**, under the **Advanced** tab.



5.5.9 **Select** – Select the **Select** button. The **Select Table** window will open.



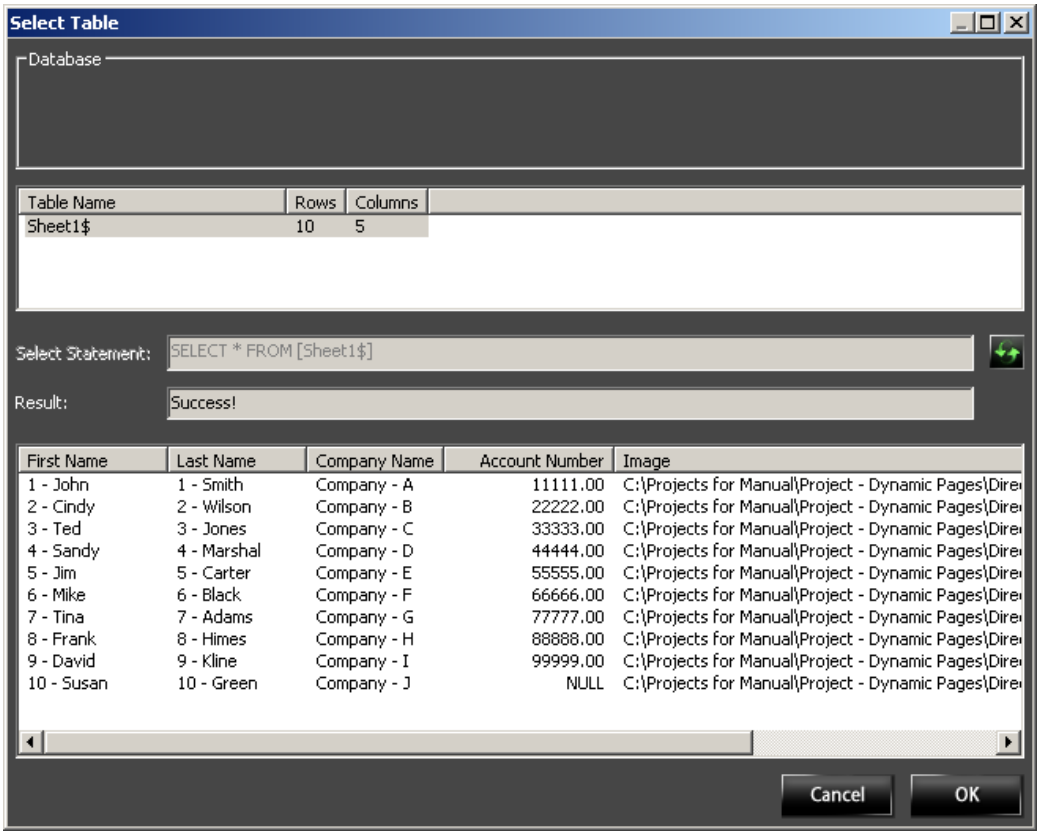
Example using Excel Spreadsheet

5.5.9.1 Select the **Table Name** that contains the content that will be used for this **Image Region.**

Note: For Excel files the **Table Name** represents the Sheets of the Excel document.

5.5.9.2 To populate the **Database** content into the viewing area, select the **Refresh** button located to the right of the **Select Statement** box.

5.5.9.3 The content of the **Table** will now be visible in the viewing area.

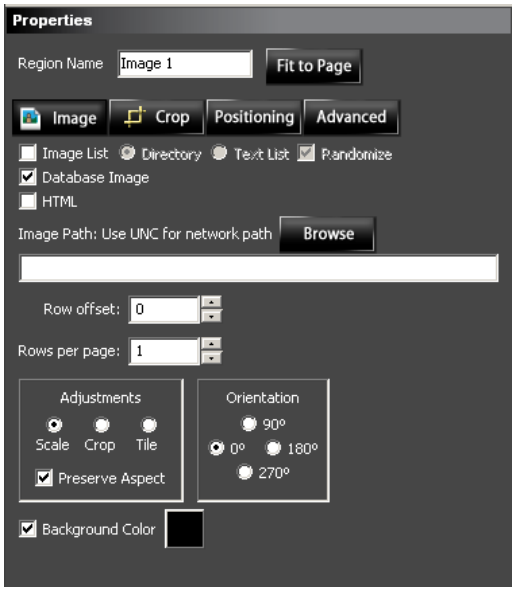


Example using Excel Spreadsheet

5.5.9.4 The **Table** is now linked to the **Region**.
Select **OK** to close the **Select Table** window.

5.5.10 Now that the **Table** is linked to the **Dynamic Region**, the **Dynamic Region** can be formatted to display specific data from the **Table**.

5.5.11 Return to the **Image** tab in the **Properties** panel.



5.5.12 **Image Path** box **-** Now that the **Image Region** is properly connected to an external **Data Source**, the **Image Path** box, can be used to point at the **Database Image** paths using a **Formatter**.
(see the **Formatter** section of this manual)

5.5.13 When setting up **Data** to be displayed in a **Dynamic Region**, there are a multitude of creative ways the **Data** can be handled and displayed.
(see the **Data Handling Options** section of this manual)

5.5.14 The **Rotations** for this **Region** will display a different row of data from the **Data Source** (as set up with **Row offset** and **Row Increment**). The number of **Rotations** will depend on how many rows of data are in the **Data Source** as well as how many other **Dynamic Regions** are on the **Dynamic Page**.
(see the **Rotations** section of this manual)

5.5.15 When developing your **Dynamic Region** the **Data** from the external **Data Source** can be inserted and viewed in the **Suite Design** window.
(see the **Refresh Now** / **Update Interval** section of this manual)

5.5.16 The content of each **Rotation** of a **Dynamic Page** can also be viewed in context of how it will be displayed with other **Pages**, by adding the **Dynamic Page** to a **Sequence**.
(see the **Adding a Dynamic Page to a Sequence** section of this manual)

6 Formatter

6.1 A **Formatter** is used to point to a cell (row and column) of a **Database**.
(see the **Noventri Suite Database Manual**)

Examples [1:2] would be row 1 column 2
[3:3] would be row 3 column 3

6.2 The **Formatter** points to a row and column in the **Spreadsheet** that contains the **Text** or path to an **Image**. This cell is the beginning point of data extraction.

6.3 All the displayed **Data** will be from the column specified, and will cycle sequentially (as set up with **Row offset** and **Row Increment**) through the rows that contain **Data**. This **Formatter** is designed to cycle through the **Data** row by row but not column by column.

Note: The first row of the Excel **Spreadsheet** is always assumed to be column headers and is therefore not accessible.

So [1:2] would actually be Excel document row 2 column 2.

6.4 When a row is selected by the **Formatter** it is recommended to always select the first row in the table that contains an entry. Using the **Row offset** option allows the first displayed entry to be selectable without the possibility of blank entries being displayed at the end.

Using the **Formatter** to select a row (other than the first row) in the table that contains an entry, will not change the number of rows to be displayed.

Example: The table being used has 10 rows of entries. However the first entry to be displayed is row 3.

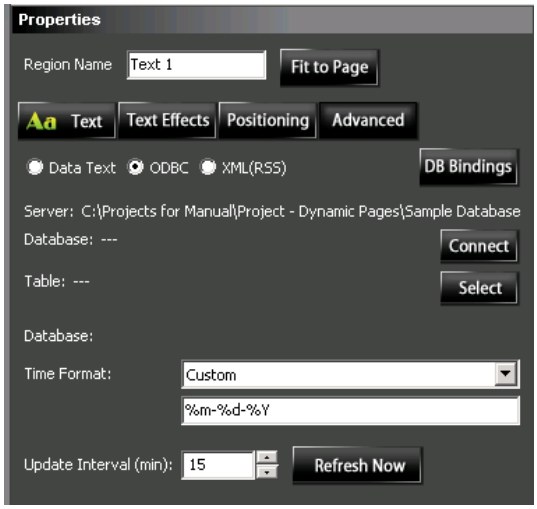
If a **Row offset** of 2 is used, row 3 will be displayed first and row 10 will be the last entry displayed. (a total of 8 **Rotations** to display these)

If the **Formatter** [3:1] is used, row 3 will be displayed first, and row 10 will be the last entry displayed. However there will be a total of 10 **Rotations** (matching the number of rows), and the last two will have this **Region** blank.

7 Refresh / Update Interval

7.1 **Refresh Now** and **Update Interval** trigger the retrieval of **Data** from the external **Data Source**.

7.2 The **Refresh Now** button and the **Update Interval** setting are both found on the **Text/Image Advanced** tab.



7.3 Refresh Now – Suite

7.3.1 The **Refresh Now** is used for updating the **Data** being displayed in **Noventri Suite**.

7.3.2 The **Refresh Now** button is only needed for retrieval of **Data** from a **Database/Spreadsheet Data Source**. Other **Data Sources**, such as **Text Lists**, auomatically refesh.

7.3.3 As long as a valid **Formatter** is entered, and the **Region** is properly linked to the **Data Source**, selecting the **Refresh Now** button will allow the **Noventri Suite** view of the **Region** to display the **Data** from the **Table**.

Note: Selecting the **Refresh Now** button only has to be done within the **Suite** the first time and then will automatically **Refresh**. It will however have to be selected again if the **Project** is closed and re-opened in the **Suite**.

7.4 Update Interval – Server

7.4.1 The **Update Interval** controls the time (in minuites) specified for how often the external **Data Source** is accessed by the **Server** and checked for updates.

Note: Setting this interval to less than 15 min may over tax the **Player** and is not recommended.

8 Rotations

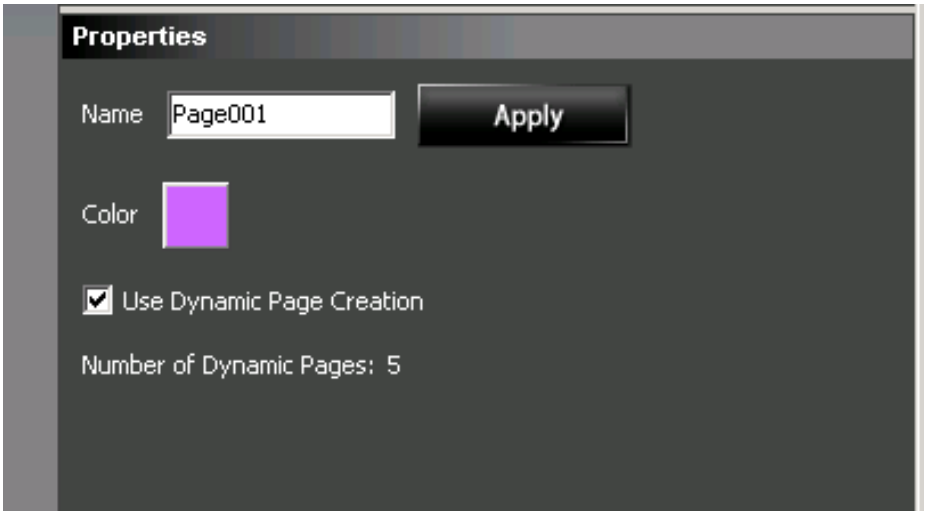
8.1 This section will help clarify how many **Rotations** will be displayed for a **Dynamic Page**.

8.2 Once a **Dynamic Page** has been set up with one or more **Dynamic Regions**, it will automatically **Rotate** the number of times needed to display all the content.

8.3 The **Region** that has the largest number of entries to be displayed will determine the number of **Rotations** that will be displayed.

8.4 **Regions** that have *less* entries to be displayed, than the number of **Rotations** (due to other **Regions** that have more entries) will be blank once their last entry has been displayed.

8.5 The number of **Rotations** is shown in the **Pages Properties** box (**Number of Dynamic Page Rotations**).



8.6 To view the content of each **Rotation**, the **Dynamic Page** has to be added to a **Sequence**.
(see the **Adding a Dynamic Page to a Sequence** section of this manual)

8.7 The total number of **Rotations** for a **Dynamic Page** can depend on the following...

- How many **Dynamic Text/Image Regions** are on the **Dynamic Page**.
- How many individual links each **Dynamic Text/Image Region** has to a **Data Source**. For example, one **Dynamic Region** may contain several **Formatters** pointing to different columns of data.
- How many **Text/Image** entries are in the **Data Source**.
- How the **Row offset** and **Row Increment** options are configured for each **Dynamic Text/Image Region**.

8.8 Simple examples of, how many **Rotations** will be displayed...

Example 1: **Dynamic Page** has one **Dynamic Region** with 5 entries...

Five **Rotations** will be displayed (**Rows** 1,2,3,4,and5)

Example 2: **Dynamic Page** has two **Dynamic Regions** with 5 entries each...

Five **Rotations** will be displayed (**Rows** 1,2,3,4,and 5 for each **Region**)

Example 3: **Dynamic Page** has two **Dynamic Regions,** one with 5 entries and the other with 7 entries...

Seven **Rotations** will be displayed (**Rows** 1,2,3,4, 5, blank, blank in the one **Region** and **Rows** 1,2,3,4,5,6 and 7 in the other **Region**)

8.9 Examples when the **Row offset** option is used.

- Row offset will skip the top entries.

Example 1: **Dynamic Page** has one **Dynamic Region** with 10 entries and a **Row offset** of 1...

Nine **Rotations** will be displayed (**Rows** 2-10)

Example 2: **Dynamic Page** has two **Dynamic Regions** with 10 entries each and a **Row offset** of 0 for the first **Region** and a **Row offset** of 1 for the second **Region**....

Ten **Rotations** will be displayed** (**Rows** 1-10 in the one **Region** and **Rows** 2-10, and a blank in the other **Region**)

Example 3: **Dynamic Page** has two **Dynamic Regions** with 10 entries each and a **Row offset** of 2 for the first **Region** and a **Row offset** of 3 for the second **Region**....

Eight **Rotations** will be displayed**
(**Rows** 3-10 in the one **Region** and **Rows** 4-10, and a blank in the other **Region**)

8.10 Examples when the **Row Increment (Rows per page)** option is used.

- Row Increment –allows rows to be skipped.

Example 1: **Dynamic Page** has one **Dynamic Region** with 10 entries, **Row Increment** set to 1 (default)...

Ten **Rotations** will be displayed
(**Rows** 1-10)

Example 2: **Dynamic Page** has one **Dynamic Region** with 10 entries, **Row Increment** set to 2...

Five **Rotations** will be displayed
(**Rows** 1,3,5,7,9)

Example 3: **Dynamic Page** has two **Dynamic Regions** with 10 entries each, with **Row Increment** set to 1 for the first **Region** and a **Row Increment** set to 2 for the second **Region**....

Ten **Rotations** will be displayed**
(**Rows** 1-10 in the one **Region** and **Rows** 1,3,5,7,9, and five blanks in the other **Region**)

Example 4: **Dynamic Page** has two **Dynamic Regions** with 10 entries each, with **Row Increment** set to 3 for the first **Region** and a **Row Increment** set to 5 for the second **Region**....

Four **Rotations** will be displayed**
(**Rows** 1,4, 7,10 in the one **Region** and **Rows** 1,6, blank, blank in the other **Region**)

8.11 Examples when the **Row offset** option is used with the **Row Increment** option...

Example 1: **Dynamic Page** has one **Dynamic Region** with 10 entries, a **Row offset** of 1 and **Row Increment** set to 2...

Five **Rotations** will be displayed
(**Rows** 2,4,6,8,and10)

Example 2: **Dynamic Page** has two **Dynamic Regions** with 10 entries each, a **Row offset** of 4 for the first **Region** and **Row Increment** set to 2 for the second **Region**....

Five **Rotations** will be displayed**
(**Rows** 1,2,3,4, 5, blank, blank in the one **Region** and **Rows** 1,2,3,4,5,6 and 7 in the other **Region**)

** **The greatest number of entries always takes priority .**

8.12 Examples when more than one cell on a **Spreadsheet** is pointed to in the same **Region**.

Examples below use the **"Sample Spreadsheet A** ".

Examples [1:1][1:2] would display JohnSmith

[1:1] [1:2] would display John Smith

[1:1] from [1:3] would display
John from Company – X

[1:1]\n[2:1] would display
John
Cindy

Note: \n can be used for a line return

Sample Spreadsheet A

First Name	Last Name	Company Name	Account Number
John	Smith	Company - X	123456.00
Cindy	Wilson	Company - Y	789123.00
Ted	Jones	Company - Z	456789.00

8.13 Examples of **Rotations** when more than one cell from a **Spreadsheet** is used in the same **Region**.

Examples below use the **"Sample Spreadsheet B"**.

Examples [1:1]\n[2:1]\n[3:1]
The 1st **Rotation** would display

1 - John
2 - Cindy
3 - Ted

then the next **Rotation** would display

2 - Cindy
3 - Ted
4 – Sandy

and so on, scrolling through the column

Sample Spreadsheet B

	A	B	C	D
1	First Name	Last Name	Company Name	Account Number
2	1- John	1 - Smith	Company - A	11111
3	2 - Cindy	2 - Wilson	Company - B	22222
4	3 - Ted	3 - Jones	Company - C	33333
5	4 - Sandy	4 - Marshal	Company - D	44444
6	5 - Jim	5 - Carter	Company - E	55555
7	6 - Mike	6 - Black	Company - F	66666
8	7 - Tina	7 - Adams	Company - G	77777
9	8 - Frank	8 - Himes	Company - H	88888
10	9 - David	9 - Kline	Company - I	99999
11	10 - Susan	10 - Green	Company - J	12121
12				

8.14 Examples of **Rotations** when more than one cell from a **Spreadsheet** is used in the same **Region** in combination with **Row offset**.

Examples below use the **"Sample Spreadsheet B**".

Examples [1:1]\n[2:1]\n[3:1] with Row offset of 1
The 1st **Rotation** would display

2 - Cindy
3 - Ted
4 – Sandy

then the next **Rotation** would display

3 - Ted
4 – Sandy
5 – Jim

and so on, scrolling through the column

8.15 Examples of **Rotation** when more than one cell from a **Spreadsheet** is used in the same **Region** in combination with **Row Increment**.

Examples below use the **"Sample Spreadsheet B**".

Examples [1:1]\n[2:1]\n[3:1] with Row Increment 3
The 1st **Rotation** would display

1 - John
2 - Cindy
3 - Ted

then the next **Rotation** would display

4 – Sandy
5 – Jim
6 – Mike

and so on, for the rest of the column

8.16 Examples of **Rotations** when more than one cell from a **Spreadsheet** is used in the same **Region** in combination with **Row offset** and **Row Increment**.

Examples below use the **"Sample Spreadsheet B**".

Examples [1:1]\n[2:1]\n[3:1] with **Row offset** 1 and **Row Increment** 3

The 1st **Rotation** would display

2 - Cindy
3 - Ted
4 – Sandy

then the next **Rotation** would display

5 – Jim
6 – Mike
7 - Tina

and so on, for the rest of the column

8.17 Examples of **Rotations** when more than one cell from a **Spreadsheet** is used in the same **Region,** in combination with **Row offset** and **Row Increment**, while also having multiple **Dynamic Regions** on a **Dynamic Page**.

Examples below use the **"Sample Spreadsheet B**".

Examples: 1st **Region** [1:1]\n[2:1]\n[3:1]
with **Row offset** 1 and **Row Increment** 3

2nd **Region** [1:1]\n[2:1]\n[3:1]
with **Row offset** 4 and **Row Increment** 3

The 1st **Rotation** would display

1st **Region**	2nd **Region**
2 – Cindy	5 - Jim
3 – Ted	6 - Mike
4 – Sandy	7 - Tina

then the next **Rotation** would display

1st **Region**	2nd **Region**
5 – Jim	8 - Frank
6 – Mike	9 - David
7 – Tina	10 - Susan

then the next **Rotation** would display

1st **Region**	2nd **Region**
8 – Frank	
9 – David	
10 – Susan	

Note: Whenever a cell is pointed to, that does not contain data
– The **Sequence Preview** will display it showing the formatters. e.g. [1:3]
– The Player will display it as blank.

9 Data Handling Options

9.1 When using **Dynamic Regions**, there are options that can be very helpful in the manipulation and displaying of the data.

9.2 Metacharacters

9.2.1 A **Metacharacter** is a character that has a special meaning (instead of its literal meaning).

\n can be used for a line return
****t can be used for a tab
**** can be used as a literal \

9.2.2 **Metacharacters** can be used in **Dynamic Text Regions**. They can be used with the **Text File** option or the **Database** option.

9.2.3 **Metacharacters** when used within a **Text File** will allow multiple lines of text to be viewed on the same **Rotation**.

Example 1: Text File Line 1
Text File Line 2
Text File Line 3\n Text File Line 4

1st **Rotation** will display - Text File Line 1

2nd **Rotation** will display - Text File Line 2

3rd **Rotation** will display - Text File Line 3
Text File Line 4

Example 2: Text File Line 1\tText File Line 2
Text File Line 3 \\ Text File Line 4

1st **Rotation** will display –

Text File Line 1 Text File Line 2

2nd **Rotation** will display -

Text File Line 3 \ Text File Line 4

9.2.4 **Metacharacters** can be used when the **Database** option is selected by including them in the **Text** box entry with the **Formatter/s**.

Note 1: Examples below use "**Sample Spreadsheet B"**

Note 2: Additional examples are available.
(see the **Rotations** section of this manual)

Sample Spreadsheet B

	A	B	C	D
1	First Name	Last Name	Company Name	Account Number
2	1- John	1 - Smith	Company - A	11111
3	2 - Cindy	2 - Wilson	Company - B	22222
4	3 - Ted	3 - Jones	Company - C	33333
5	4 - Sandy	4 - Marshal	Company - D	44444
6	5 - Jim	5 - Carter	Company - E	55555
7	6 - Mike	6 - Black	Company - F	66666
8	7 - Tina	7 - Adams	Company - G	77777
9	8 - Frank	8 - Himes	Company - H	88888
10	9 - David	9 - Kline	Company - I	99999
11	10 - Susan	10 - Green	Company - J	12121
12				

Example: First Name: [1:1] \t Last Name: [1:2] \n \nCompany: [1:3}

The 1st **Rotation** will display -

First Name: 1 - John Last Name: 1 – Smith

Company: Company - A

The 2nd **Rotation** will display -

First Name: 2 - Cindy Last Name: 2 – Wilson

Company: Company - B

and so on, for the rest of the columns

9.3 Variables

9.3.1 A **Variable** is a name or character that represents a value. There are two specific **Variables** set up for use in **Dynamic Text Regions**.

%page_current	will display the current **Dynamic Page Rotation** number.
%page_total -	will display the total number of **Dynamic Page Rotations** for that **Region**.

Example:

Dynamic Region\n Rotation number %page_current of %page_total

The 1st **Rotation** will display -

Dynamic Region
Rotation Number 1 of 10

The 2nd **Rotation** will display -

Dynamic Region
Rotation Number 1 of 10

and so on, for the rest of the **Rotations**

9.3.2 When using the **Database** option for a **Dynamic Text/Image Region** the column name may be used as a **Variable** for the column number in the **Formatter**.

This has two advantages...

- Seeing the column name instead of the column number makes it easier to create the **Formatters** and recognize them.

- If columns are added or deleted from the Database, causing the column numbers to change, the **Formatter** will still point to the correct column.

Example:

[1:Company Name] would be the same as [1:3]

9.4 Row offset

9.4.1 The **Row offset** option allows the top line/s of a **Text File, Image List,** or **Table** to be ignored.

Example 1: If there are 5 rows of data in the **Text/Image List/Table** and the **Row offset** is set to 1, then only rows 2-5 will cause a **Rotation**, therefore prompting only four **Rotations** of the **Dynamic Page**.

Example 2: If there are 5 rows of data in the **Text/Image List/Table**. If **Row offset** is set to 2, two rows will be skipped, so only rows 3 - 5 will cause a **Rotation**.

9.5 Row Increment (Rows per page)

9.5.1 The **Row Increment (Rows per page)** option allows rows to be skipped.

Example: Leaving the **Row Increment** set to 1, will display every row.
(unless already skipped due to **Row offset**).

Example: If **Row Increment** is set to 2, every other row will be displayed, and likewise if it is set to 3, every third row, and so on.

9.5.2 Changing the **Row Increment** to 0 will display *only* the 1st row (after taking into account the **Row offset**). This will cause the **Region** to function as a normal **Region** (not **Dynamic**).

Note: The 1st row (after any **Row offset**) will always be shown, then the rows being skipped will follow. So, if the **Row Increment** is set to 2, the 1st row will be displayed first and 3rd row second, etc...

10 Dynamic Page in a Sequence

10.1 The **Dynamic Page** needs to be added to a **Sequence** for two reasons.

First of all - While creating a **Dynamic Page Project**, it is convenient to see and cycle through the **Rotations** that will be created, as well as set the length of time each **Rotation** will be displayed. This will enable the content to be seen ahead of time, just as it will be displayed when the **Project** is **Assigned to a Player**.

Second – The **Sequence** must be created before the **Project** can be **Published**.

10.2 The **Dynamic Page** should be added to a **Sequence** as any other **Page** would be.
(see the **Noventri Design** Manual)

10.3 Once a **Dynamic Page** is in a **Sequence,** it's colored block in the **Timeline** will say **"Dynamic"**.

10.4 Normally when a **Page** is added to the **Timeline**, the width of the colored block is adjusted to determine the length of time that the **Page** will be displayed. For a **Dynamic Page** the length of time that is selected is the length of time that each **Rotation** will be displayed.

Example: If the length of time that was selected for the **Dynamic Page** colored block is 1 min. and it is formatted to **Rotate** 20 times, each one of the 20 **Rotations** will be displayed for 1 min. So a total of 20min before the next colored block (**Page**) in the **Sequence** will be displayed.

Note: The **Page Duration** indicated to the right of the **Timeline** does not take into account the number of **Dynamic Page Rotations**.

10.5 If a **Transition** is added to the **Dynamic Page** it will be the **Transition** used between each created **Page**.

10.6 To cycle through the **Rotations** that will be created and view them in the **Preview Panel**, select the **Dynamic Page** colored block in the **Timeline** and cycle through the **Rotations** by using the right and left arrow keys on the keyboard.

10.7 Once a **Project** with a **Dynamic Page** is completed it can be **Published, Scheduled,** and **Assigned** to a **Player** as any other **Project** would be.
(see the **Noventri Design, Schedule,** and **Manage** Manuals)

NOVENTRI
301-790-0103 | www.noventri.com
20940 Twin Springs Dr., Smithsburg, MD 21783 USA