DYNAMIC PAGES

What are Dynamic Pages?

As a tool, the Dynamic Pages function can simplify the creation of the display of large amounts of information. It can also simplify the creation of the display of information that is frequently changed. The function enables this by allowing a page to be populated and repopulated with data that is stored outside of the page itself, creating or deleting pages as necessary to display the information.

In some respects, dynamic pages are to the Noventri Suite what a form letter would be to a word processor. Instead of creating page after page that, but for a few changes to a text region and/or an image region, would be nearly identical to one another, you could create one page that would be formatted to accommodate those changes. Using the Dynamic Pages function, the Noventri Suite would then ***automatically*** create or delete pages so that only those needed to display the new information exist.

Example uses of the Dynamic Pages function:

e.g. 1 – A sequence of pages acts as the workplace's display calendar, cycling through a series of upcoming events. The user creates only one page containing a text region. When played, that region is formatted to be populated with several calendar events per page, spawning a new page until there are no remaining calendar events in the source material, before repeating the cycle. While new information for upcoming months will need to be entered into the source document, the new pages needed to accommodate this information can be as detailed as the previous ones without having to be created manually.

e.g. 2 – Linked to the appropriate database, a schedule of airline flights complete with accompanying logos and up-to-the-minute flight times populates the corresponding text and image regions of a dynamic page layout. Pages are automatically created as an increasing number of flights are scheduled. Pages are automatically deleted as a decreasing number of flights are scheduled.

e.g. 3 – A manufacturing warehouse allocates one of their displays for the use of inventory. This display screen shows a listing of various items in the inventory accompanied by a picture of the item and the corresponding quantity of such items currently in stock. Because the display is linked to the database of inventory items, the display is updated whenever the database is updated. The employee(s) responsible for inventory can simply change the quantities entered in the database and/or remove items from the inventory if they are no longer carried, without having to create or delete pages.

… and there are many more! Use of Dynamic Pages can often be tailored to your individual needs.

*The Dynamic Pages function is an advanced feature of the Noventri Suite. Completion of this section of the manual assumes a preexisting knowledge of project creation, basic design features, and terminology previously covered.

Creating Dynamic Pages

Creating a Source Document

Because the Dynamic Pages function relies on an external source of information to populate its regions, a source document should be created first. A source document is a file in which you will store the information that you eventually want your Dynamic Pages to extract and display. For ease of use, it is recommended that you use a spreadsheet or other database program to create your source document.

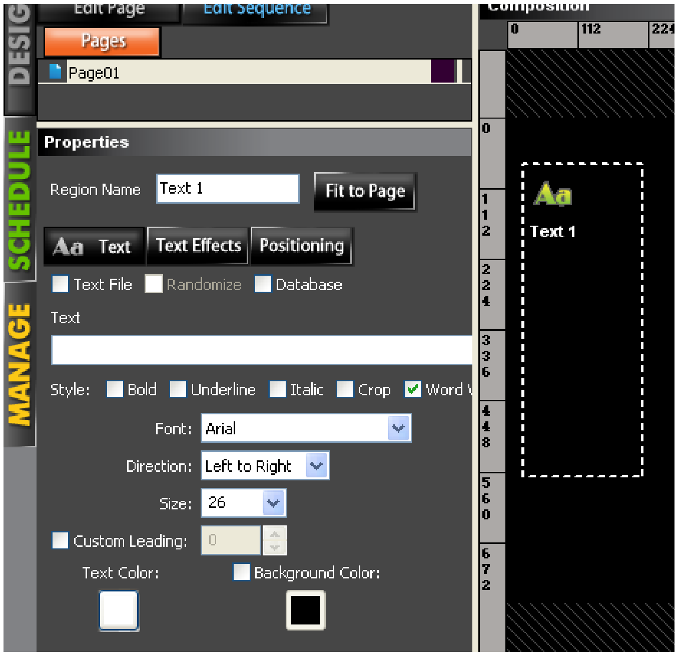
The Dynamic Pages function will extract information from your designated spreadsheet in a sequential order by *column*, with the next item to be displayed extracted from the following *row*. Beginning with the second row of the spreadsheet, therefore, enter information into columns in the order that you would wish for them appear on your pages.

NOTE: The cells in the first row of a spreadsheet will not be extracted from for purposes of filling out your page's regions. Information entered into these cells will instead be considered as column headers.

Linking the Source Document

Linking a source document designates that particular document as the source your Dynamic Page will extract data from. This section will guide you through the process of linking a source document to the page you are currently working on.

In the design tab of the Noventri Suite, click to select the text region you intend to populate with information from your source document, in order to display the text 'Properties' panel.



Region Name: Text 1

□ Text File □ Randomize □ Database

Text

Style: □ Bold □ Underline □ Italic □ Crop ☑ Word \

Font: Arial

Direction: Left to Right

Size: 26

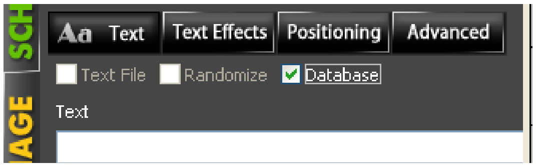
□ Custom Leading: 0

Text Color: □ Background Color:

□ ***Database* –** If your source document was constructed and saved using a database program, click this checkbox. Selecting this option will make available the ‘Advanced’ tab.

□ ***Text File* –** If your source document was constructed and saved using a text program, click this checkbox. Selecting this option will allow you to browse for your text file.

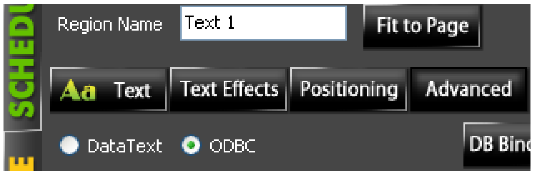
Click the ‘Advanced’ tab.



□ Text File □ Randomize ☑ Database

Text

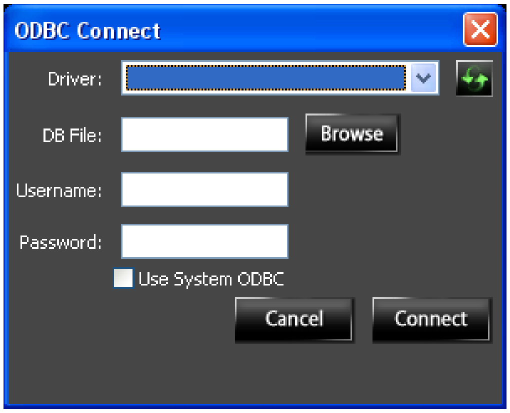
Click to select **ODBC.**



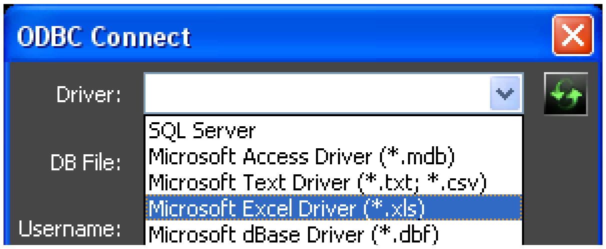
Region Name: Text 1

○ DataText ● ODBC

Click the ‘Connect’ button to open the ‘ODBC connect’ window.



In the ‘ODBC Connect’ window, select the appropriate driver from the drop-down menu, making sure that the file extension (.xls, for example) matches the file extension of the source document you intend to use.


ODBC Connect
Driver:
DB File:
Username:
SQL Server
Microsoft Access Driver (*.mdb)
Microsoft Text Driver (*.txt; *.csv)
Microsoft Excel Driver (*.xls)
Microsoft dBase Driver (*.dbf)

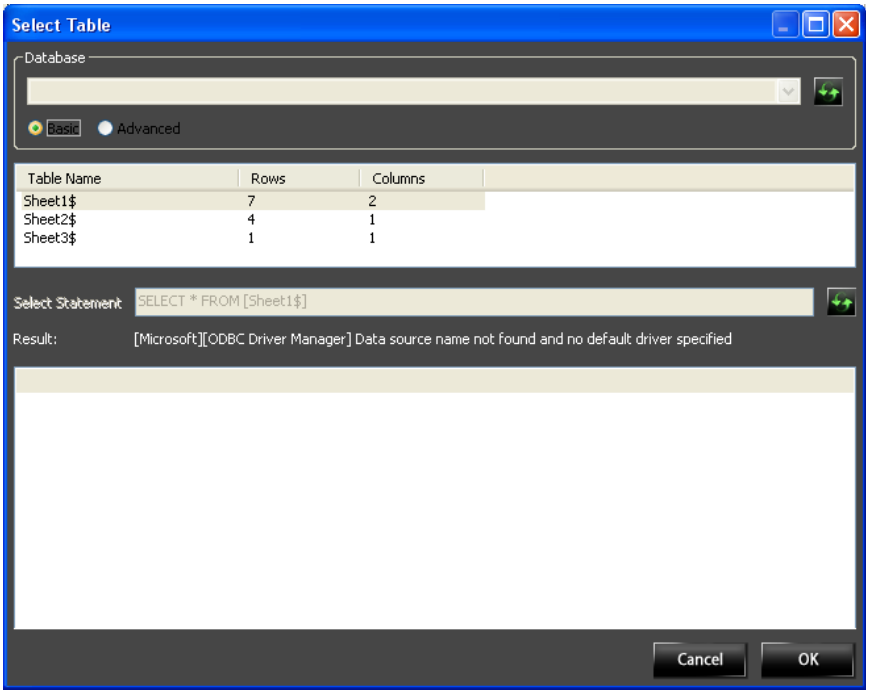
NOTE: If your source document does not match any of the extensions in the drop-down menu, you must click on **File > Save As** within the source document, and save the file with an extension that would match one of the drop-down menu options.

After selecting your file extension from the drop-down menu in the ‘ODBC Connect’ window, click Browse to search for the database file you intend to use as your source document.

NOTE: Before clicking ‘Connect,’ make sure your spreadsheet is closed.

Click the ‘Connect’ button to close the ‘ODBC Connect’ window and link the source document to your page.

In the ‘Advanced’ tab, click the ‘Select’ button to open the ‘Select Table’ window.
By default, the first table in your spreadsheet is the selected table. Click on a different table to select it, if desired. A selected table will be highlighted.



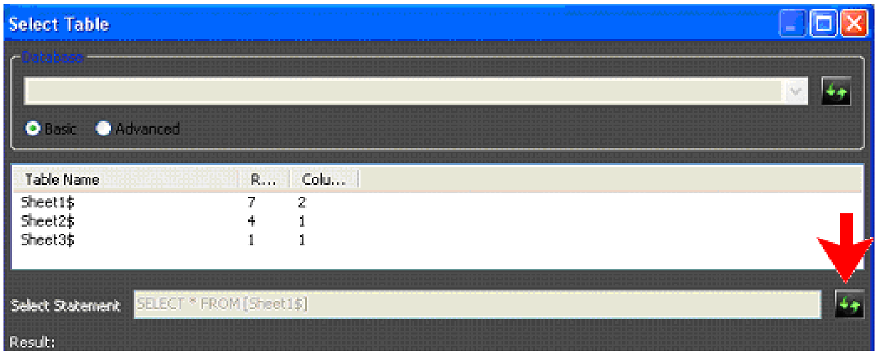
● Basic ○ Advanced

Table Name	Rows	Columns
Sheet1$	7	2
Sheet2$	4	1
Sheet3$	1	1

Select Statement: SELECT * FROM [Sheet1$]

Result: [Microsoft][ODBC Driver Manager] Data source name not found and no default driver specified

After selecting which table of information you would like to use to populate your page's regions, click the second 'Refresh' button which will be located next to the 'Select Statement' bar. Doing so, will copy the information from your spreadsheet and place it in the 'Select Table' window for your overview.

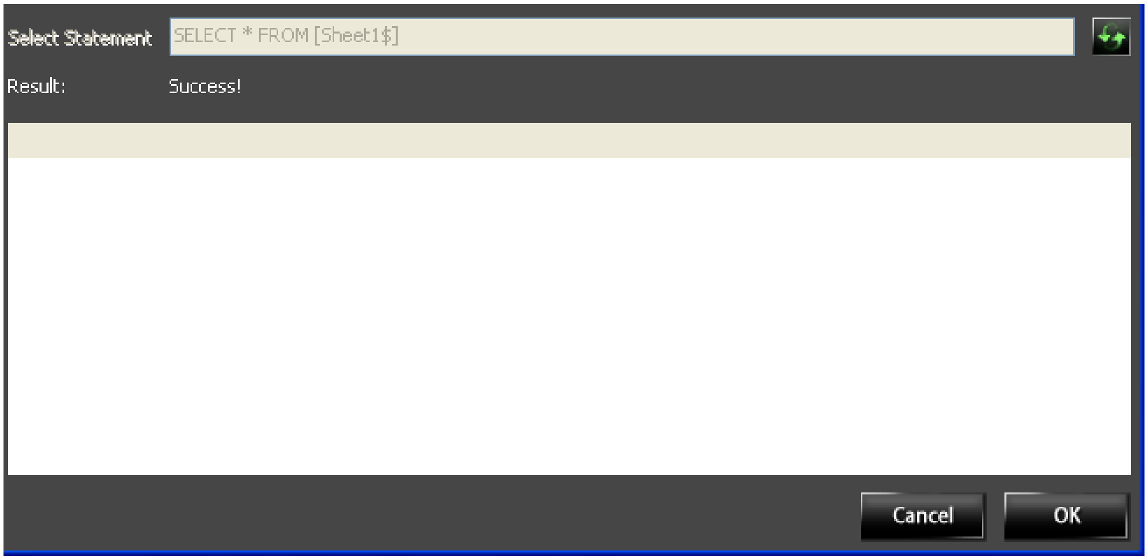


● Basic ○ Advanced

Table Name	R...	Colu...
Sheet1$	7	2
Sheet2$	4	1
Sheet3$	1	1

Select Statement: SELECT * FROM [Sheet1$]

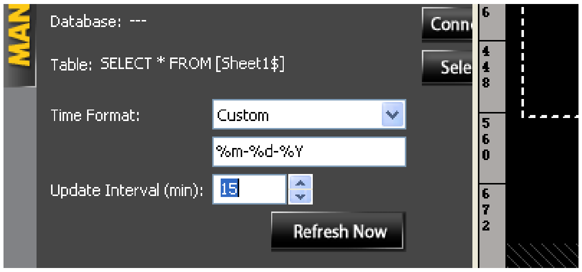
Click the 'OK' button to confirm the selection and close the 'Select Table' window.



Select Statement SELECT * FROM [Sheet1$]

Result: Success!

'Update Interval' sets how often, in minutes, you would like the program to check the source document and update, in order to display any changes that may have been made. The default interval is set to update once every fifteen minutes. This frequency can be changed by clicking the up or down arrows.



Database: ---

Table: SELECT * FROM [Sheet1$]

Time Format: Custom

%m-%d-%Y

Update Interval (min): 15

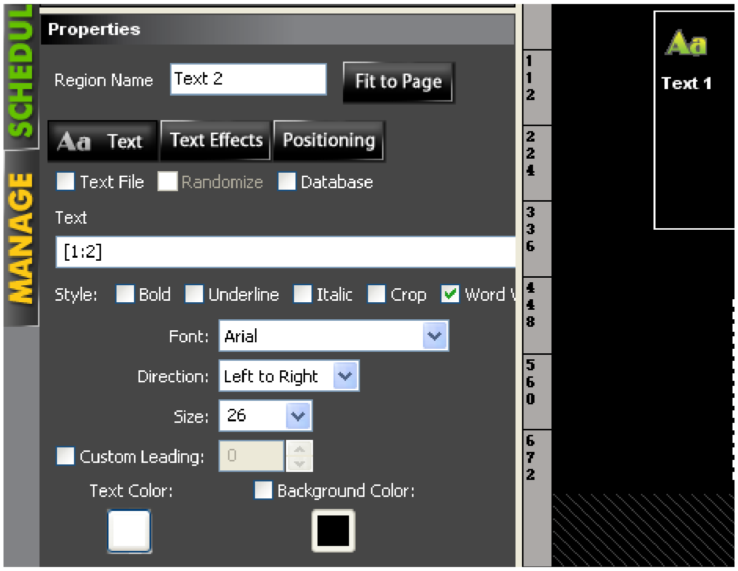
After completing this process, you will have successfully linked your source document to your project, though further commands will still be required in order to refine your project and cause the data to display in your regions. To learn about this process and other commands, please continue on to the next section.

Displaying Source Document Data

Now that you have linked a source document to your project, the regions on your Dynamic Page where you would display data from the source document will need to be formatted using a query language. A query language is a programming language designed to manage and manipulate database functions and information. In a designated manner, commands using this query format will extract information from the database in order to display it onscreen. This section will guide you in the use of these functions.

NOTE: The Dynamic Pages function may create several pages to accommodate the display of your data. Though not all media on a page need to adhere to the query format, any images or bodies of text that have not been formatted in such a way will remain unchanged from page to page, as more Dynamic Pages are created.

To begin, click on a text region and select the ‘Text’ tab.



Region Name: Text 2

□ Text File □ Randomize □ Database

Text: [1:2]

Style: □ Bold □ Underline □ Italic □ Crop ☑ Word W

Font: Arial

Direction: Left to Right

Size: 26

□ Custom Leading: 0

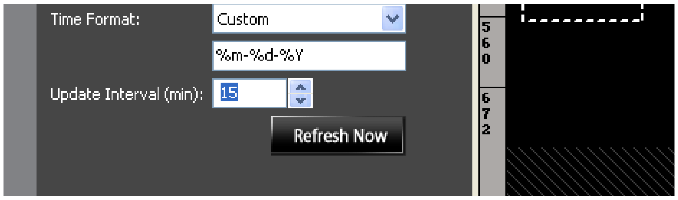
Text Color: □ Background Color: ■

Any text typed into the text box will display in the text region. Here, it may be useful to imagine each cell of your spreadsheet as having an address. To retrieve information from a cell, the address must be entered into the text box using the following format: [1:2]

The closed brackets indicate, to the Dynamic Page function, the spreadsheet cell itself. In the above example of a formatter, the 1 indicates the row number in which the cell is located. In the above example of a formatter, the 2 indicates the column in which the cell is located. These two numbers are separated, for reference, by a colon. Thus, taken all together, entering [1:2] into the text box would extract whatever information occupies the cell that is in the first row of the second column, causing it to appear at the corresponding location within the text region.

NOTE: The cells in the very first row of a spreadsheet will not be extracted from for purposes of filling out your page’s regions. Information entered into these cells will instead be considered as column headers. Thus, for example, the formatter [1:2] actually references the cell located in the second row and second column of your spreadsheet. The formatter, [2:2], references the cell located in the third row and second column of your spreadsheet, and so on. The address of each cell will thus be naturally offset by one row.

After entering the appropriate formatters into the text box, click the ‘Advanced’ tab to display the ‘Refresh Now’ button.



Time Format: Custom

%m-%d-%Y

Update Interval (min): 15

Clicking the 'Refresh Now' button will cause the program to extract the corresponding information from your spreadsheet and place it where your formatters are, thus populating your region with the information. Once the 'Refresh Now' button has been pressed once, the region will automatically refresh as new text, commands, and/or formatters are entered into the text box. This eliminates the need to manually refresh your page's regions as you continue to update and design it.

NOTE: This function will be active only as long as your project remains open. To disable this function, you must close your project. By default, this function will not be active when you reopen your project.

Additional Formatters

In addition to the formatters that reference a database cell, the Design Suite can make use of other formatters in order to arrange information within a region. The following are commonly used text formatters and their functions:

\n – indicates to the region a use of the enter/return key. This formatter can be typed in manually, but will be entered automatically if the enter/return key is pressed while the cursor is in the text box.

\t – indicates to the region a use of the tab key.

%page_current – causes the current page number to display at a corresponding location in the region. This formatter must be entered manually and will display once the 'Use Dynamic Page Creation' checkbox has been checked and the project has been refreshed (see section X.X).

%page_total – causes the total number of Dynamic Pages to display at a corresponding location in the region. This formatter must be entered manually and will display once the 'Use Dynamic Page Creation' checkbox has been checked and the project has been refreshed (see section X.X).

NOTE: %page_current and %page_total are often used in a single text region together, separated by reference language. For example: '*%page_current out of %page_total*,' can be placed into its own text region and would read 2 out of 7 when the second of seven Dynamic Pages is displayed, and so on.

These formatters can be placed into the text box by themselves, or you can build additional text around them. Keep in mind that when the page is refreshed, all formatters that have been entered correctly will not appear as they do in the text box, but rather, will display corresponding information from your source document in your regions. For example, a text box may read: *There are\n [1:2]\n[1:1]\n remaining in stock.*

Depending on its dimensions, your text region could then display as:

There are
23
#4533 oval gaskets
remaining in stock.

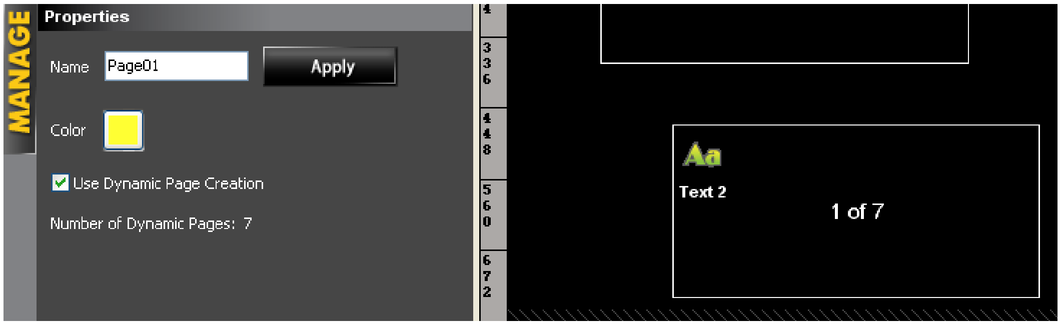
You could infer from the above example that '23' was a quantity entered into one cell of the user's spreadsheet, while '#4533 oval gaskets' serves as an item description that was entered into a different cell.

Dynamic Page Creation

With the previous steps complete, you will now be able to preview how many Dynamic Pages will need to be created in order to display the contents of your database. To various degrees, you will also be able to limit and manipulate how much of your database is displayed. These previously unavailable functions are made available by selecting the Dynamic Page Creation tool and will be explored in this section.

To continue, deselect the text region by clicking in the background of the page.

In the 'Properties' panel, click the 'Use Dynamic Page Creation' checkbox. If the 'Refresh Now' button located in the 'Advanced' tab has been pressed, this will display the number of Dynamic Pages that will be created so that all designated information in the database you have linked can be displayed.



Name: Page01

☑ Use Dynamic Page Creation

Number of Dynamic Pages: 7

Now, during the display of your sequence, the Dynamic Page function will continue to populate your regions with sequential entries from your source document. These entries will appear in the locations designated by your formatters.

When using Dynamic Page Creation, you do not need to change the *row* number of your formatter in order to display an entire column's contents. In order to display the contents of a different *column*, you must either change which column is designated in the formatter or create an additional formatter. The formatters are designed to cycle through cells from row to row, but not from column to column. Dynamic Page Creation will automatically create new pages, filling them in with the next entry found in the designated column.

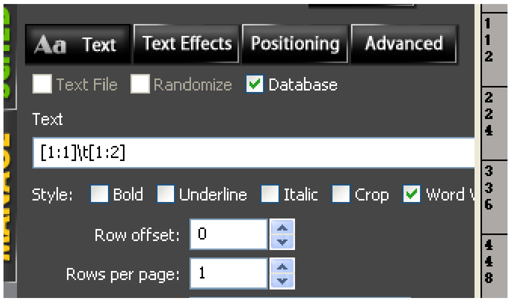
To illustrate:

> Entered into the text box, [1:1] will display the first non-header item in the first column. The next page created will display the information from [2:1] at the same location where [1:1] displayed on the previous page. The third page created will display the information from [3:1] at the same location where [1:1] displayed on the first page and so on.
>
> Dynamic Page Creation will continue creating and populating pages in this pattern until the last entry in the column designated by the formatter has been displayed.

NOTE: Cells in a designated column, other than those in the first row or those that follow the last entry in the column, that remain empty will still be extracted from. In the sequence preview, these cells will display in the text region as if the formatter had not been refreshed. In the actual display of your sequence, these cells will appear as a blank space where the formatter would have been.

<u>Manipulating the Display of Data</u>

In addition to displaying the number of Dynamic Pages that will be created, clicking the 'Use Dynamic Page Creation' checkbox will make available the 'Row offset' and 'Rows per page' functions. These features are located in the 'Text' tab and may have a direct bearing on the number of Dynamic Pages that will be created.



□ Text File □ Randomize ☑ Database

Text

[1:1]\t[1:2]

Style: □ Bold □ Underline □ Italic □ Crop ☑ Word \

Row offset: 0

Rows per page: 1

The 'Row offset' and 'Rows per page' functions can be used in conjunction with one another or independent of one another allowing you to further manipulate the display of your database.

Row offset **–** Using the up and down arrows, this function allows you to designate a different row as the point from which all formatters in the region will begin to extract data, effectively (but not actually) changing the cell designated by a given formatter.

To illustrate:

> Normally, entering [1:1] into the text box will display the first non-header item in the first column. The next page created will display the information from [2:1] at the same location where [1:1] displayed on the previous page. The third page created will display the information from [3:1] at the same location where [1:1] displayed on the first page and so on.
>
> Wanting your display to ignore the first four rows of data, you change the 'Row offset' to 4. This indicates that, for the first Dynamic Page created, each formatter in the region will extract data from a cell four rows lower in the table than the formatter would otherwise indicate. With 'Row offset' set to 4, then, entering [1:1] into the text box will display the fifth non-header item in the first column. The next page created will display the information from [6:1] (rather than [2:1]) at the same location where [5:1] displayed on the previous page. The third page created will display the information from [7:1] (rather than [3:1]) at the same location where [5:1] displayed on the first page and so on. Each formatter in the region is thus offset by the indicated number of rows.
>
> Dynamic Page Creation will continue creating and populating pages in this pattern until the last entry in the column designated by the formatter has been displayed. Once the last entry in the designated column has been displayed, the player will simply proceed to the next page in the sequence, rather than cycle back to the beginning of the column to create and display more pages using the unextracted data located in the offset portion of the column.
>
> ***NOTE:*** Oftentimes, your page will make use of a formatter that is not identical to another formatter on the same page. Under these circumstances, it is possible (depending on the 'Rows per page' you have set and the number of blank cells to be found within your database) that on a given Dynamic Page, one or more of your formatters will reference a blank cell. When this occurs, a blank cell will still be extracted from. In the sequence preview, this cell will display in the text region as if the formatter had not been refreshed. In the actual display of your sequence, this cell will appear as a blank space where the formatter would have been.

When in use, this function will reduce the amount of data extracted which may, in turn, reduce the amount of Dynamic Pages created. You can see if there has been a change to the number of Dynamic Pages by viewing the total located under the 'Use Dynamic Page Creation' checkbox.

The ‘Row offset’ function is useful to those who wish to eliminate from the display a section of data that may no longer be relevant. For instance, a calendar display may be offset so as not to show events/scheduling for dates/times that have already past.

Rows per page – Using the up and down arrows, this function allows you to designate that data be extracted only from every nth row. This will cause each formatter to skip a select number of rows in your database before extracting the data that is to be displayed on subsequent created pages.

To illustrate:

> Normally, entering [1:1] into the text box will display the first non-header item in the first column. The next page created will display the information from [2:1] at the same location where [1:1] displayed on the previous page. The third page created will display the information from [3:1] at the same location where [1:1] displayed on the first page and so on.
>
> Wanting your formatters to display only every third item in their respective columns, you change the ‘Rows per page’ to 3. This indicates that, for each Dynamic Page created, each formatter in the region will extract data from a cell three rows lower in the table than what was extracted by the formatter on the previous page. Thus, with ‘Rows per page’ set to 3, entering [1:1] into the text box will display the first non-header item in the first column. The next page created will display the information from [4:1] at the same location where [1:1] displayed on the previous page. The third page created will display the information from [7:1] at the same location where [1:1] displayed on the first page and so on. Notice that each formatter in the region is not displaying three rows worth of data. Rather, the ‘Rows per page’ function allows you to apply a new pattern to the extraction and display of database information.
>
> Dynamic Page Creation will continue creating and populating pages in this pattern until the last entry in the column designated by the formatter has been displayed. Once the last entry in the designated column has been displayed, the player will simply proceed to the next page in the sequence, rather than cycle back to the beginning of the column to create and display more pages using the unextracted data located in the rows that were skipped.

When in use, this function will reduce the amount of data extracted which may, in turn, reduce the amount of Dynamic Pages created. You can see if there has been a change to the number of Dynamic Pages by viewing the total located under the ‘Use Dynamic Page Creation’ checkbox.

The ‘Rows per page’ function is similar to the ‘Row offset’ function in that it is useful to those who wish to eliminate from the display a section of data that may no longer be relevant. It does this by selecting for cells based on repeatable patterns. For instance, a calendar display may be set to skip a certain number of rows so as to only display events/scheduling for the upcoming weekends of the month, while ignoring events scheduled on other days, for purposes of the display.

The ‘Rows per page’ function reaches even greater potential when it is used with formatters that are placed in tandem. Then, this function can be used to display several rows worth of data on a single page without doing so redundantly.

To illustrate:

> You decide that displaying ten rows of database content per page would be more efficient, while still remaining legible on the display screen. So, in the text box, you arrange the formatters for the first ten cells in sequential order separated by the formatter for a use of the enter/return key ([1:1]\n[2:1]\n[3:1]\n, etc.). The first page would display ten cells worth of data much as it appears in your database.
>
> Then, wanting your display to avoid skipping or repeating any database content, you set the ‘Rows per page’ to 10. The next page created would display the next ten cells in the same manner, effectively showing ten new rows per page and so on until Dynamic Page creation has displayed the rest of the column.

Using ‘Row offset’ and ‘Rows per page’ in Conjunction – A database need not be created with these functions in mind in order to be displayed efficiently. Oftentimes, these functions are used together, complimenting one another as they help direct the display of an existing database. Moreover, these functions can achieve much greater potential within your display when they are used together in conjunction with multiple regions.

To illustrate:

> You decide that displaying ten rows of database content per page would be more efficient. However, in order to remain legible on the display screen, these ten rows must be displayed in two columns. To facilitate these two columns, two separate text regions are created (one for each column).
>
> In the text box for the first region, you arrange the formatters for the first five cells in sequential order separated by the formatter for a use of the enter/return key ([1:1]\n[2:1]\n[3:1]\n, etc.). For this region, ‘Row offset’ remains at 0 while you change ‘Rows per page’ to 5, accommodating for the five rows worth of data referenced here.
>
> Into the text box for the second region, you copy the same formatters. Again, you change ‘Rows per page’ to 5, to accommodate for the five rows worth of data referenced here. Wanting this region to display the data that sequentially follows that which is displayed in the first region, you change the ‘Row offset’ to 5, so

that the second region will always show data that is located 5 rows lower than the data that is shown in the first region.

Your first page will now display rows 1-5 of a column in the first region and cells 6-10 of a column in the second region. The next page that is created will display rows 6-10 of the column in the first region and rows 11-15 of the column in the second region. The third page created will display rows 11-15 of the column in the first region and rows 16-20 of the column in the second region and so on until the rest of the column's contents have been displayed.

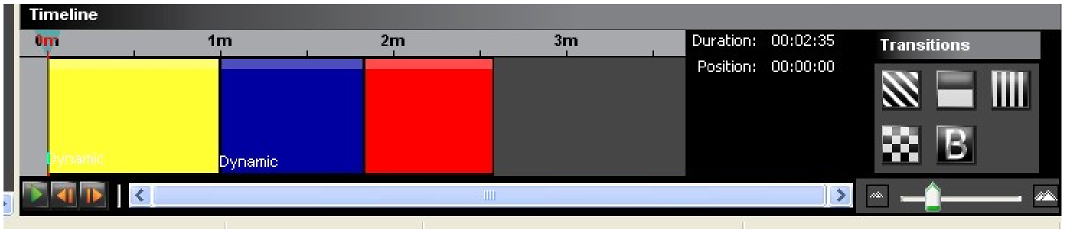
The effect that this particular manipulation creates is that of the data being 'marched' across the display screen. This 'march' also allows for a difference between the perceived display speed of the data and the actual display speed of the page, as information that first appears in the second region will seem to linger in the first region before giving way to the next body of data.

After adjusting the settings to account for the display of your data, you will be ready to incorporate your Dynamic Pages into a sequence. To learn about this process, please continue on to the next section.

Adding Dynamic Pages to a Sequence

Now that you have adjusted your formatters and other settings to account for the display of your data, you are ready to incorporate your Dynamic Pages into a sequence. Dynamic Pages display in a manner different from other pages when incorporated into a sequence. This section will familiarize you with how Dynamic Pages will function when placed in a sequence.

In the 'Edit Sequence' tab, Dynamic Pages can be added to a sequence in the same way as other pages – by clicking on the page thumbnail in the 'Pages' sub-panel. (See section 10.5.2) Pages placed into a sequence will have a label indicating if the page is 'Dynamic'.



Duration: 00:02:35
Position: 00:00:00

For sequencing purposes, all of the Dynamic Pages that would be created from a single database will be represented as only one color block on the sequence timeline. When the sequence is displayed, however, this section of the timeline will queue the display of the first page and each subsequent page that would be created to display your database. On the scale, the length of the color block indicates the length of time that each individual page created by the Dynamic Pages function will be displayed.

To illustrate:

> You add a Dynamic Page to a sequence that, when displayed, would create five pages in order to display all of the designated data. Wanting each created page to remain on the display screen for ten seconds, you adjust the color bar on the timeline to ten seconds, *not* 50 seconds.

NOTE: The sequence duration indicated to the right of the timeline does not take into account the number of dynamic pages that will be created when determining the duration of your sequence display.

Although all of the Dynamic Pages that would be created from a single database are represented as only one color block on the sequence timeline, each individual page can be previewed in the sequence preview window. Dynamic Pages can be previewed by clicking on the appropriate color box in the timeline to select it. Next, press the right arrow key to scroll forward through the pages and the left arrow key to scroll backward through the pages. As you do so, you can see each page that will be created appear in the preview window. When playing, after the last entry in the database has been displayed, the display will continue on to the next page placed in your sequence.

With the previous steps complete, your project is now ready to be previewed on your monitor. (See section 11.)