View Pointers

Take advantage of CGI and HTML to customize web views and eliminate useless links.

By Keith Reichley

HIGHLIGHTS

$$ViewTemplate

CGI

HTML

Well-designed Notes views shine in their ability to present dynamic information on your Domino-based web site. As documents are added to the database, they immediately fall into place, without the need to create static HTML links.

Views can be even more effective when they're customized to provide sounder navigation and a more intuitive interface. This article highlights several techniques developers can use to enhance the quality of any site using Notes views. Each of these techniques, unless noted otherwise, works in versions of Notes/Domino 4.6 and earlier.

Standard view template

Experienced Domino developers know $$ViewTemplate forms let them customize the appearance of web views. A $$ViewTemplate form must contain an editable text field called $$ViewBody. This field determines where on a page Domino displays the contents of your view. Beyond that, $$ViewTemplate forms can contain whatever text, images, links, etc., needed to give a view the desired look-and-feel. For more information about $$ViewTemplate forms, refer to the *Domino Users Guide* available from Lotus Development at http://www.lotus.com/.

When a Notes view is published to a browser, Domino looks for a form called $$ViewTemplate for *<viewname>*, where *<viewname>* is the view's alias (or its name in the absence of an alias). For example, the form named $$ViewTemplate for States associates that form with the view called States. If there isn't a form called $$ViewTemplate for *<viewname>*, Domino looks for the more generic $$ViewTemplateDefault. If that form doesn't exist, Domino displays the view with its own standard appearance.

This default appearance leaves something to be desired, especially in its navigation bar that contains icon and text links for Previous, Next, Expand, and Collapse. This navigation bar appears both above and below the body of the view.

For most Notes views, these standard links are necessary. However—and this is a common criticism—at least one of these links is always irrelevant. For example, when a view is opened by the browser, the Previous link is superfluous. There is no previous page of the view to which the visitor can return. The link is there, but nothing happens if the visitor clicks on it. Another example: When a categorized and collapsed view is opened, the Collapsed link is redundant. Useless links like these tend to confuse the user. They fly in the face of good web design.

You can do better for your web site customers. Here's a technique for building a better view template that dynamically provides relevant view navigation links.

Better default view template

The key to providing meaningful links for view navigation is knowing where the visitor is

Keith Reichley, a technology consultant for a large insurance company, has been in IT for eight years—the last four as a Notes and Domino developer. Most recently, he has focused on Intranet/Internet development. His numerous articles and presentations at national and regional technical seminars have contributed little to the college funds of his children, Spencer and Emma.

Table 1: **Hidden fields for a better view template**—Add these nine hidden fields to the top of your view template form. These fields are the driving force behind dynamic view navigation links.

Field Name:	Purpose:	Type:	Formula:
Path_Info	Captures URL of current page.	Editable	none
HTTP_Referer	Captures URL of previous page.	Editable	none
start	Captures the value of URL Start argument if it exists.	Computed	`start2 := @Middle(Path_Info; "Start="; 2); start1 := @If(@Contains(start2; "&"); 1; @If(@Contains(start2; "."); 3; 2)); @If(@Contains(@UpperCase(Path_Info) ; "START="); @TextToNumber(@Middle(@UpperCase(Path_Info); "START="; start1)); 0)`
count	Captures value of URL Count argument if it exists.	Computed	`@If(@Contains (@UpperCase(Path_Info) ; "COUNT="); @TextToNumber(@Middle(@UpperCase(Path_Info); "COUNT=";2)); 0)`
newView	Determines if the visitor is at the beginning page of a view.	Computed	`right8 := @UpperCase(@Right(Path_Info; 8)); @If(right8 = "OPENVIEW" | @Contains(right8; ".NSF") | (right8 = "PANDVIEW" & start = 1) | start = 1; "Yes"; "No")`
midView	Determines that the visitor isn't at the beginning page of a view.	Computed	`wasOne := @If(@Contains(@UpperCase(HTTP_Referer); "START=1"); "Yes"; "No"); isOverOne := @If(start > 1; "Yes"; "No"); @if(wasOne = "Yes" | isOverOne = "Yes"; "No")`
endView	Determines that the visitor clicked on the Next link on the last page of the view and the same page was redisplayed.	Computed	`lastpage := @UpperCase(@Right(HTTP_Referer; 15)); thispage := @UpperCase(@Right(Path_Info; 15)); @If(lastpage = thispage; "Yes"; "No")`
expandedView	Determines if the view has been partially or fully expanded.	Computed	`@If(@Contains(@UpperCase(Path_Info) ; "EXPAND"); "Yes"; "No")`
collapsedView	Determines if the view has been partially or fully collapsed.	Computed	`@If(@Contains(@UpperCase(Path_Info) ; "COLLAPSE"); "Yes"; "No")`

in a view. Is the visitor on the first page in the view? On the last page? Somewhere in the middle? Is the view expanded or is it collapsed?

The answers to these questions can nearly always be found by examining the values of Path_Info and HTTP_Referer. These are two of the many CGI variables supported by Domino. CGI variables are standard methods web servers use to gather user information: user name, URL of current and previously visited pages, browser being used, etc.

Armed with the values of Path_Info and HTTP_Referer, you can use the power of the Hide-When text property to display the good view navigation links and to hide the links that don't apply.

If you already have a form called $$ViewTemplateDefault, make a copy of it and rename the old one. If you don't have a default view template, create a new form called $$ViewTemplateDefault. Give the form one editable text field called $$ViewBody. Next, add the nine hidden Text fields shown in table 1 to the top of the form. Finally, add the navigation links to the form. Where the standard view template puts all of the links on one always-visible line, you should place your links individually and in well-placed pairs on separate lines. Some links should be positioned above the $$ViewBody field, while others belong below it.

If you're building your view template from scratch, table 2 shows the formulas for the individual view navigation links.

Table 2: **Link by link**—Formulas for each navigation link.

Link:	Action Hotspot Formula:
Previous	`@DbCommand("Domino"; "ViewPreviousPage")`
Next	`@DbCommand("Domino"; "ViewNextPage")`
Expand All	`@Command([ViewExpandAll])`
Collapse All	`@Command([ViewCollapseAll])`

Navigation links above the view

Certain view navigation links make the most sense when they are placed above the body of the view. Figure 1 shows how these look on the form in design mode.

Continued

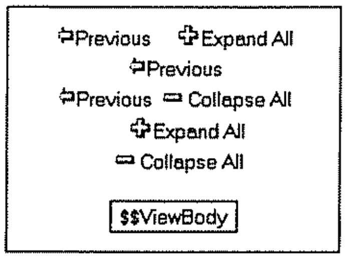


Figure 1: **Navigation links above the view**–These links and link pairs enhance the visitor's experience when placed above the view.

Set these links above the $$ViewBody field on your form. The Hide-When formulas that refer to the hidden fields at the top of the form ensure only relevant links are displayed (table 3).

Navigation links below the view

Now that the above the view navigation links are in place, create the links that belong below the body of the view. Figure 2 shows how these appear on the form in design mode.

Enter these link sets below the $$ViewBody field on your form. As with the links above the view, the Hide-When formulas ensure that only relevant links are displayed to the visitor (table 4).

Exceptions

Even as powerful as Domino is in delivering browser-ready applications, it can't give us everything. Until Lotus comes out with the @Command([ReadMyMind])—or the ESP function in LotusScript—you need to be aware of two exceptions:

1. **Next link:** The only time you can be sure if a page is the last in the view is if the previous page's URL (HTTP_Referer) is the same as the current page (Path_Info). This call is made in the formula for the endView field (table 1). This technique conservatively displays the Next link if it's at all possible the displayed view page may not be the last.
2. **Expanded/Collapsed:** The only sure way for a browser to know if a view is expanded or collapsed is if the URL contains the words expand or collapse. To be safe, this

Table 3: **Formulas for links above the view**—Hide-When formulas make sure only the relevant links are displayed.

Link Set:	Hide Paragraph When Formula Is True:
Previous	`check := expandedView = "No" & collapsedView = "No" & newView = "No"; !check`
Previous | Expand All	`check := expandedView = "Yes" & newView = "No"; !check`
Previous | Collapse All	`check := expandedView = "Yes" & newView = "No"; !check`
Expand All	`check := newView = "Yes" & collapsedView = "Yes"; !check`
Collapse All	`check := newView = "Yes" & expandedView = "Yes"; !check`

Now....Stampede *TurboGold*® more than doubles the speed of Lotus Notes™ database replication.

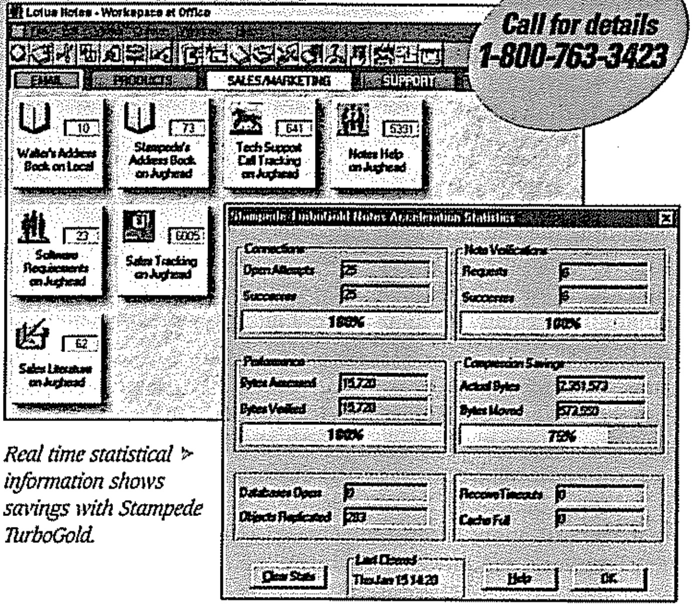
Stampede Technologies' patented acceleration technology speeds the replication of Lotus Notes databases with these unique speed-boosting innovations:

- **Replication accelerator**—compression of Notes database updates into a continuous stream of data.
- **Intelligent caching**—automatic downloading of only data changed since last access.
- **Passive replication**—automatic caching of Lotus Notes databases on *TurboGold* Verifiers closest to the end-user.
- **Data file verification**—time and date stamp matching eliminates unnecessary transfers.

And, the benefits don't stop there. Stampede *TurboGold* also accelerates Lotus cc:Mail (DB6 & DB8) and file based applications such as Lotus Approach, Microsoft Access, Microsoft Office, etc.

Don't take our word for it—see for yourself.
Evaluation software is available. Call today for details.

Business Partner


Lotus Notes - Workspace at Office
EMAIL
PRODUCTS
SALES/MARKETING
SUPPORT
Walter's Address Book on Local
Stampede's Address Book on Jughead
Tech Support Call Tracking on Jughead
Notes Help on Jughead
Software Requirements on Jughead
Sales Tracking on Jughead
Sales Literature on Jughead
Stampede TurboGold Notes Acceleration Statistics
Connections
Open Attempts
Successes
Note Verifications
Requests
Successes
Performance
Bytes Assessed
Bytes Verified
Compression Savings
Actual Bytes
Bytes Moved
Databases Open
Objects Replicated
Receive Timeouts
Cache Full
Last Cleared
Clear Stats
Help
OK
Call for details 1-800-763-3423

Real time statistical ▸ information shows savings with Stampede TurboGold.

Stampede Technologies, Inc., 65 Rhoads Center Drive, Dayton, OH 45458
Tel: 937-291-5035, Toll free: 800-763-3423, Telfax: 937-291-5040, Web: www.stampede.com

Stampede TurboGold U.S. Patent #5,682,514 Stampede TurboGold is a registered trademark of Stampede Technologies. All other product names are the property of their respective owners.

Table 4: **Formulas for links below the view**—Again, the Hide-When formulas makes sure only relevant links display.

Link Set:	Hide Paragraph When Formula Is True:
Next	`check := (newView = "Yes" | midView = "Yes") & expandedView = "No" & collapsedView = "No" & endView = "No"; !check`
Next | Expand All	`check := (newView = "Yes" | midView = "Yes") & collapsedView = "Yes" & endView = "No"; !check`
Next | Collapse All	`check := (newView = "Yes" | midView = "Yes") & expandedView = "Yes" & endView = "No"; !check`

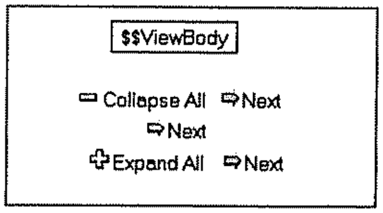


Figure 2: **Navigation links below the view**—These links and link pairs are most effective when placed below the view.

technique assumes that, in the absence of these words in the URL, the view is neither expanded nor collapsed. If the view being displayed by the browser isn't categorized, all is well. If the view is categorized, once the visitor clicks on a twistie to either expand or collapse a category, the appropriate Expand All or Collapse All link kicks in.

Modify this technique to meet the requirements of your web site. For example, if you're creating a view template form specifically for a categorized and collapsed view, you may want to change the Hide-When formula for any of the lines containing the Expand All link so the link isn't hidden when the view is first displayed. The power is yours.

More view enhancements

A recurring request from our clients has been to build sites in Notes with easily maintainable content, but that look like other sites on the Web. Our team often exceeds these expectations by combining the power of Notes views with embedded HTML to achieve some rather non-Notes-like results.

Create a table-like view

When Domino renders a view to the browser, it actually creates an HTML table. To make your view look like a standard HTML bordered and link-less table, put embedded HTML to work.

Continued

FETCH YOUR E-MAIL FOR JUST $99 A MONTH

The whole idea behind Internet e-mail is speed and simplicity. So why is it so complicated to get it up and running in a network environment?

With dotNet, it isn't.

We've made it so simple that any old dog can learn this trick: Internet e-mail delivered directly through your current version of Microsoft Exchange, Novell GroupWise, Lotus Notes, cc:Mail, MS Mail, or MHS-based system. With seamless addresses too: yourname@yourcompany.com.

And with dotNet, you won't have to beg. dotNet service starts at just $99 per month.

The solution is simple. Call dotOne at **1-800-826-4666** today. Call anybody else and you're barking up the wrong tree!

dotOne
One Source. Total Messaging.

© 1997 dotOne Corporation. dotNet is a registered trademark of dotOne Corporation. All product names mentioned are the property of their respective holders. Visit our web site as you guessed it—www.dotone.com

dotNet.

With dotNet, you can affordably send and receive Internet e-mail messages seamlessly from within these and all MHS e-mail packages.

Lotus Notes
Lotus cc:Mail
Microsoft Exchange Server
Microsoft Mail
Novell GroupWise
WordPerfect Office

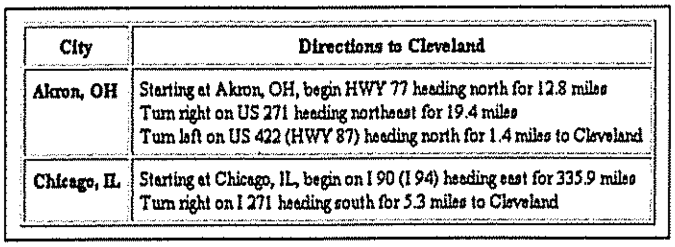
Now Includes faxing from within E-MAIL!

Start Today
1.800.826.4666
http://www.dotone.com

To achieve the borders, include this HTML in the title of the first column in front of any desired title text:

```
[</table><table cellpadding=5 border=1><tr><th>]
```

The cellpadding is optional. Figure 3 shows the result of this HTML applied to a simple two-column view.



City	Directions to Cleveland
Akron, OH	Starting at Akron, OH, begin HWY 77 heading north for 12.8 miles Turn right on US 271 heading northeast for 19.4 miles Turn left on US 422 (HWY 87) heading north for 1.4 miles to Cleveland
Chicago, IL	Starting at Chicago, IL, begin on I 90 (I 94) heading east for 335.9 miles Turn right on I 271 heading south for 5.3 miles to Cleveland

Figure 3: **Table-like view**—This view contains embedded HTML that produces cell borders and suppresses column links.

You'll also notice in figure 3 that the view has no links. Sometimes the view says it all and a link isn't needed. Unless you tell it not to, Domino always makes the left-most column into a link. If the left-most column is categorized, you get the link in the form of a twistie. If the column isn't categorized, the value of the column becomes the link.

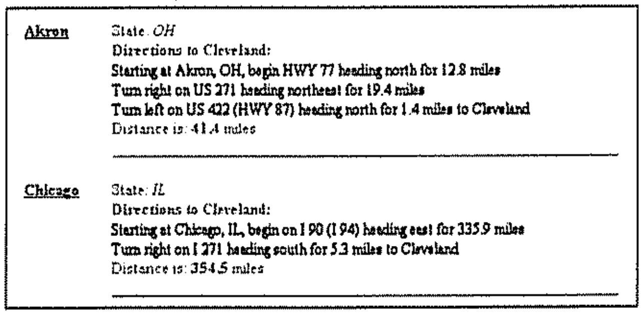
How do you tell Domino not to create the link? In 4.6, it's easy. Double-click on the column to bring up the Column Properties InfoBox. Click on the beanie tab and de-select the option labeled Show Values in this column as links.

In 4.5 and older, just add this HTML to the beginning of the column's formula to achieve the same results:

```
"[</a>]" +
```

Stacked and colorful view

One of our customers wanted a view that didn't look like a view. She wanted to display, in an attractive way, a variety of information without forcing the visitor to scroll horizontally. Figure 4 shows a view that stacks text and field values for a more readable user experience.



Akron
State: *OH*
Directions to Cleveland:
Starting at Akron, OH, begin HWY 77 heading north for 12.8 miles
Turn right on US 271 heading northeast for 19.4 miles
Turn left on US 422 (HWY 87) heading north for 1.4 miles to Cleveland
Distance is: 41.4 miles

Chicago
State: *IL*
Directions to Cleveland:
Starting at Chicago, IL, begin on I 90 (I 94) heading east for 335.9 miles
Turn right on I 271 heading south for 5.3 miles to Cleveland
Distance is: 354.5 miles

Figure 4: **More interesting page**—This page looks more like a standard HTML page than it does a Notes view. Creative use of HTML is the key.

To build a page like this, start with a simple two-column view. Make sure that the view's properties allow for multiple lines by selecting at least 2 for the Line per row option on the Styles tab.

Next, add the following HTML tags, field references, and static text to the column formulas.

For column 1:

```
"[<b>]" + City + "[</b>]"
```

For column 2:

```
"[<font size=3 color=\"#669999\">]State:<i>] " + State +
"[</I><font size=3> color=\"#999999\"><br>
<b>Directions to Cleveland:</b><br>
<font size=3 color=\"#000000\">]" + Directions + "[<font
size=2 color=\"#669999\"><br>
Distance is: <b>]" + Distance + "[</b> miles<hr><br>]"
```

Product *News*

Find and Replace for Developers

TeamStudio Configurator is a global search and replace tool for Lotus Notes applications. Developers can find all occurrences of a piece of text or field name in a database design and optionally replace them with another string. Configurator integrates completely with Notes and can be accessed through a SmartIcon from the design pane. US$300 per developer.

Ives Development, Inc.
30 Doaks Lane
Marblehead, MA 01945
(781)639-8779, fax (781)639-1422
http://www.teamstudio.com/

Head-Start Scripting

Mental Notes ScriptCoach is a Notes database that contains a library of tested and ready-to-use LotusScript functions and examples for developers to copy and paste into their applications. For beginners, every script has a walk-through section that gives a line-by-line explanation of how the code works. ScriptCoach also offers a version control feature for managing reusable code. US$695 for a 1-5 user license and US$99 for a 1-year subscription for upgrades, US$995 for a 6-10 user license and US$149 for a 1-year subscription for upgrades, US$1,295 for a site license and US$249 for a 1-year subscription for upgrades.

Mental Notes
110 E 31st Street, 4th Floor
New York, NY 10016
(212)545-1241, fax (212)545-1586
http://www.mentalnotesinc.com/

Project Management

InfoImage Team Project 5.1 is an optionally web-enabled project management system that runs in the Notes environment. This collaborative environment allows management of subprojects by individual team members across multiple sites and resource management across multiple projects. The latest version offers a new end-user navigator, an author-based security model, project templates, and tight integration with Microsoft Project 98. US$7,000 per server (with volume discounts available), US$1,400 for web-enabling module.

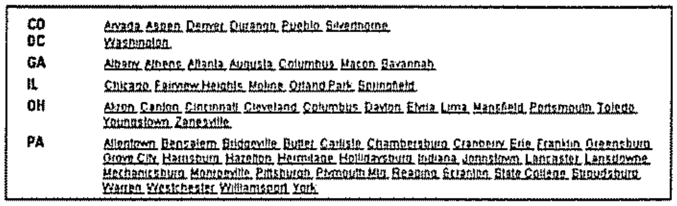
InfoImage, Inc.
100 West Claredon, Suite 2310
Phoenix, AZ 85013
(602)234-6900, fax (602)234-6950
http://www.infoimage.com/

Judicious use of break tags and font tags with color and size options overrides the color-impaired and column-bound defaults of the standard Notes view.

Real customizing power: HTML views

Notes 4.6 gives web site developers a wealth of powerful new design features. One of the most impressive lets Domino render a view as pure HTML to the browser.

Figure 5 shows a two-column categorized view. Look, ma! No twisties! To save screen real estate, I've displayed the cities for each state category—not one above the other but in long rows. If you know HTML, HTML views are for you. For more information on creating twistie-less views, see Kevin Pauli's article, "A Cure for the Common View," in the August 1998 issue of LOTUS NOTES & DOMINO ADVISOR.



CO	Arvada Aspen Denver Durango Pueblo Silverthorne
DC	Washington
GA	Albany Athens Atlanta Augusta Columbus Macon Savannah
IL	Chicago Fairview Heights Moline Orland Park Springfield
OH	Akron Canton Cincinnati Cleveland Columbus Dayton Elyria Lima Mansfield Portsmouth Toledo Youngstown Zanesville
PA	Allentown Bensalem Bridgeville Butler Carlisle Chambersburg Cranberry Erie Franklin Greensburg Grove City Harrisburg Hazelton Hermitage Hollidaysburg Indiana Johnstown Lancaster Lansdowne Mechanicsburg Monroeville Pittsburgh Plymouth Mtg Reading Scranton State College Stroudsburg Warren Westchester Williamsport York

Figure 5: **HTML view**—What's more impressive? A categorized view without twisties, or column values displayed horizontally rather than vertically?

The first step is to open the View Properties InfoBox, click on the beanie tab, and select Treat View contents as HTML.

Next, add these HTML tags and field references to the column titles and formulas.

Column 1 title:

```
<table width="100%">
```

Column 1 formula:

```
"</tr><tr></tr><tr align=left valign=top></td>
<td width=\"15%\"><font size=5><b>" + State + "</b></td>
<td width=\"85%\">"
```

Column 2 formula:

```
viewname := "yourviewname";
path1 := @ReplaceSubstring(@Subset(@DbName; -1);"\\";"/");
path2 := "<font size=2><a href=\"/" + path1 + "/" +
   viewname + "/" + @Text(@DocumentUniqueID) +
   "?OpenDocument\">" + City + " </a>";
"&nbsp" + path2
```

The extra `</tr><tr>` pair in the Column 1 formula adds an extra line break between the state categories.

Summary

Creatively combining the power of native Notes design with embedded HTML and CGI variables lets you make the most of Notes views. Do your content managers a favor by using views as much as possible. Exceed their expectations by shaping the views to the look and feel they want. ■

...a suite of Domino applications to manage knowledge

getting knowledge to who needs it,
when it is needed

Network Delivered Knowledge

leverage Notes and work the Web...really

accessed from a central Catalog with its
knowledge-mining KnowBots

knowledge is managed in:
- on-line courses
- books, policies, documentation
- job aids
- events and headlines
- knowledge databases

the check list:
- ✓ common look-and-feel throughout the suite
- ✓ outliner for document creation and management
- ✓ document access logging (who, when, how long)
- ✓ document version control with rollback
- ✓ authorship assignments, scheduled revisions
- ✓ subscriber security model
- ✓ boilerplate library for document assembly
- ✓ form-based ACL management
- ✓ economical per-server and corporate licensing
- ✓ open source code version available

www.coursewaresource.com

The $$ Forms Used in Web Applications

For information on using some of these forms, see Chapter 10.

Form Name	Requires the Field	Remarks
`$$ViewTemplateDefault`	$$ViewBody	Acts as the default form (template) for views on the Web.
`$$ViewTemplate` for *viewname*	$$Viewbody	Where *viewname* is the alias (preferred) or view name, serves as the form (template) used to display that view.
`$$NavigatorTemplateDefault`	$$NavigatorBody	Serves as the default form (template) used to display navigators.
`$$NavigatorTemplate` for *navigatorname*	$$NavigatorBody	Where *navigatorname* is the navigator name, serves as the default form (template) used to display that navigator.
`$$SearchTemplateDefault`	$$ViewBody	Serves as the default form (template) for search results on the Web.
`$$SearchTemplateDefault` for *viewname*	$$ViewBody	Where *viewname* is the view name, serves as the default form (template) for search results for a view.
`$$Search(alias)`		If you create a form with the alias $$Search, Domino will use it whenever a Web user initiates a search on a Domino server. If you create no such form, Domino uses the file search.htm in the icons directory.

appendix B

$$ Web Forms and Fields

B

The following are special Domino-reserved fields used in Web applications.

Field name	Value
`$$ViewBody`	"ViewName" or a formula that resolves to "ViewName"
`$$ViewList`	None
`$$NavigatorBody`	"NavigatorName" or a formula $$ that resolves to "NavigatorName"
`$$Return`	A text message or a formula that computes a message a user sees when he submits a form
`$$HTMLHead`	A formula that computes the Head tag information
`$$QueryOpenAgent`	"AgentName" or a formula that resolves to "AgentName"
`$$QuerySaveAgent`	"AgentName" or a formula that resolves to "AgentName"

$$Return Document Deleted

Using CGI variables to capture user info automatically

CGI programs are a standard for interfacing external applications with HTTP servers. When Web users submit forms, the HTTP server gathers the information and passes it on to a CGI program.

Although Domino obviates the need for many types of CGI programs that are needed with a typical HTTP server, Notes application developers still have the opportunity to capture cgi variables in applications. Include any of the fields below in your form to capture Common Gateway Interface (CGI) environment variables. These fields let you capture information such as the IP (Internet Protocol) address of the user submitting the form. Be sure to mark the fields "Hide when Editing," so that users cannot enter information in them.

Field Name	Returns
Remote_Host	The hostname making the request.
Remote_Addr	The IP address of the remote host making the request.
Remote_User	Authentication method that returns the username they have authenticated as.
HTTP_User_Agent	The browser the client is using to send the request.
HTTP_Referer	The URL of the page the user used to get here.
Server_Software	The name and version of the information server software running the CGI program.
Server_Name	The server's hostname, DNS alias, or IP address as it would appear in self-referencing URLs.
Server_URL	
Gateway_Interface	The version of the CGI spec to which the server complies.
Server_Protocol	The name and revision of the information protocol this request came in with.
Server_Port	The port to which the request was sent.

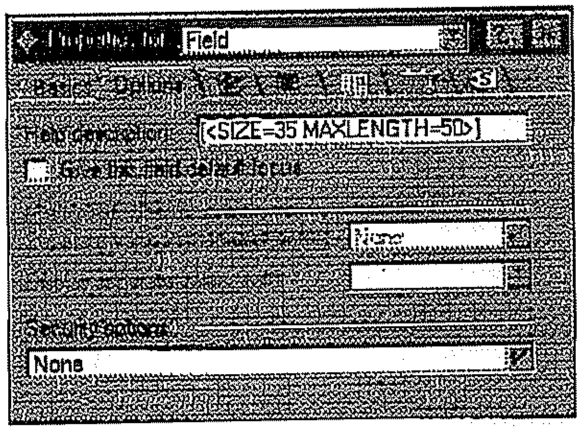
Request_Method	The method with which the request was made. For HTTP, this is "GET," "HEAD," "POST," etc.
Path_Info	The extra path information, as given by the client. In other words, scripts can be accessed by their virtual pathname, followed by extra information at the end of this path. The extra information is sent as PATH_INFO.
Path_Translated	The server provides a translated version of PATH_INFO, which takes the path and does any virtual-to-physical mapping to it.
Script_Name	A virtual path to the script being executed, used for self-referencing URLs.
Query_String	The information which follows the ? in the URL which referenced this script.
Auth_Type	If the server supports user authentication, and the script is protected, this is the protocol-specific authentication method used to validate the user.
Remote_Ident	This variable will be set to the remote user name retrieved from the server. Usage of this variable should be limited to logging only.
Content_Type	For queries which have attached information, such as HTTP POST and PUT, this is the content type of the data.
Content_Length	The length of the content as given by the client.
HTTP_Accept	The MIME types which the client will accept, as given by HTTP headers
HTTPS	

For more information, see the CGI Environment Variables Specification at http://hoohoo.ncsa.uiuc.edu/cgi/env.html.

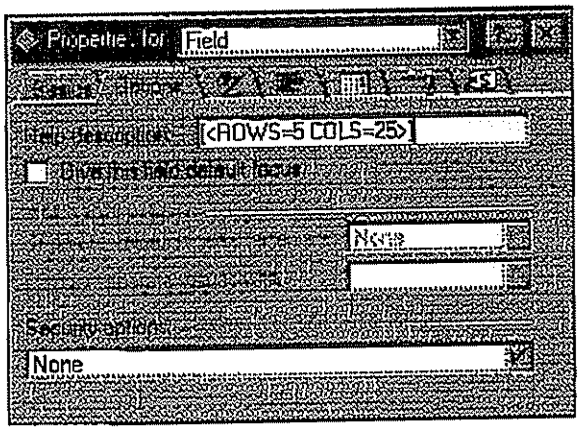
Adding HTML attributes to editable fields

Domino lets you control the size and length of fields in the form. To override the field defaults, enter HTML code in the Help description box of the Field Properties dialog box. For information on writing HTML code, visit the HTML Spec Web site at http://www.w3.org/hypertext/WWW/MarkUp/html-spec/html-spec_toc.html.

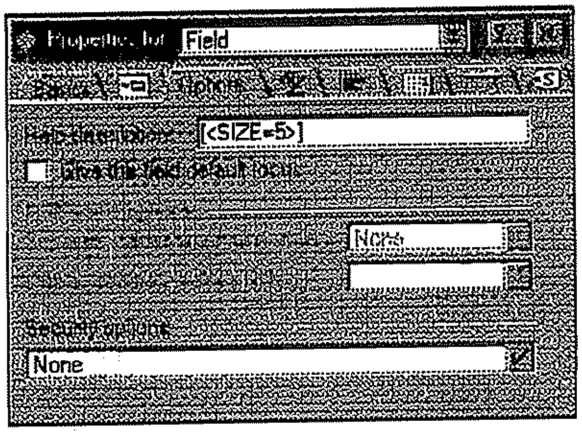
Specifying the size and maximum length of text fields
You can specify the size and length of text fields. For example, the figure below specifies that the Name field display 35 characters, and accept a maximum of 50 characters.


Field
Help description: <SIZE=35 MAXLENGTH=50>
None
None

Specifying the number of rows and columns of a text box in rich text fields
You can specify the size of the text box in rich text fields. For example, the figure below specifies that the Comments field be 5 rows long and 25 columns wide.


Field
Help description: <ROWS=5 COLS=25>
None
None

Specifying the maximum number of visible choices in keywords fields
You can specify the maximum number of rows that are visible in keywords fields. For example, the figure below specifies that the Request field has 5 rows. If more than 5 keywords exist for that field, a scroll bar will be used to scroll to other keywords.



Properties for Field

Help description: [<SIZE=5>]

Give this field default focus

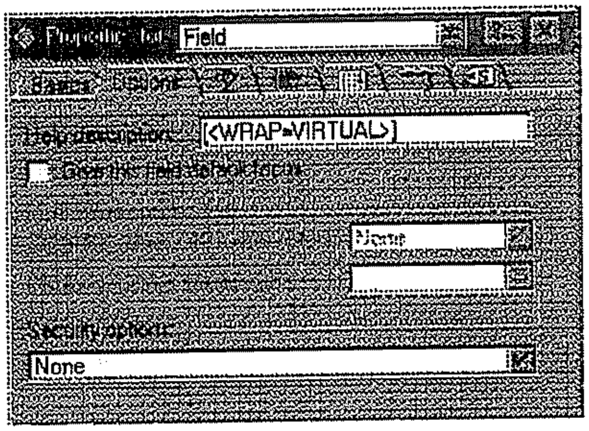
None

Security options

None

Specifying the wrap setting for a rich text field

You can specify the wrap setting for a rich text field. For example, HTML tag WRAP=VIRTUAL in the figure below specifies that text entered in the Comments field wrap, and that no line feeds or carriage returns be inserted at the end of lines.



Properties for Field

Help description: [<WRAP=VIRTUAL>]

Give this field default focus

None

Security options

None