Notes.net

Iris Today

Using Notes to customize QuickPlace

by
Julie
Forgo

Level: Intermediate
Works with: Designer 5.0
Updated: 12/08/1999

Without a doubt, Domino/Notes is the heavy-hitter of groupware development environments. But there are times when you don't need the muscle of a Domino application to get the job done. Enter QuickPlace, a Lotus solution that lets you rapidly build a team workspace that you can deploy on a corporate intranet, extranet, or on the Web. In as little as five minutes, using only an Internet browser, you can build and deploy a useful and secure electronic forum.

QuickPlace is quickly catching on as a powerful way to create and manage a shared workspace for broadcasting information to large groups of users, or for allowing small teams to share their work. Hundreds of Domino/Notes enterprise customers are evaluating QuickPlace as a tool for deploying collaborative applications that do not need the full power of a Domino application.

For example, Lars Borjessen is a Lotus Business Partner who uses a QuickPlace to introduce himself and showcase his work for prospective customers. He is also integrating QuickPlace applications in his Domino/Notes solutions. A supply chain system he designed for suppliers to the automotive industry uses Domino applications for benchmarking production and quality figures, for maintaining information about suppliers, and for facilitating communication between suppliers. A QuickPlace application provides an online discussion forum for the suppliers, with rooms for general discussion and inner rooms for more focused discussions. Lars feels that if QuickPlace had been available earlier, he would have used it for everything except for the benchmarking subsystem. He does not view QuickPlace as a competitor in the custom solution business, but as a powerful tool for providing a quick and satisfying solution for his customers. He has customized QuickPlace applications for customers to created very focused electronic forums.

What can Domino designers do with QuickPlace?

If QuickPlace is so easy to use, does it squeeze application developers out of the picture? Not at all. Domino/Notes business partners have added QuickPlace to their bag of tricks. QuickPlace makes an excellent tool for demonstrating the immediate benefit of collaborative software to customers. Customers who start with QuickPlace can later transition to fuller-featured Domino applications. And application developers have found that they can use their Domino/Notes experience to add value to applications that include QuickPlace components, especially with the customization tools available in the **QuickPlace Development Kit (QDK)**.

In this article, you'll see some examples of how you can use Domino/Notes, along with the QDK, to further customize your QuickPlace applications.

QuickPlace, Notes, and Domino

To understand how QuickPlace fits into the Domino/Notes picture, let's take a closer look at how this product is put together. QuickPlace is not Notes Lite -- that is, it is not a stripped-down version of Notes. The product was designed to be completely independent of Domino/Notes. You can run a QuickPlace server as an overlay on a Domino server, but you can also install a QuickPlace server as a standalone server, independent of Domino. QuickPlace does, however, take advantage of the Notes object model, which gives an advantage to Domino application developers who want to extend the capabilities of QuickPlace with customization. Also, the QuickPlace server was developed by extending the capabilities of the Domino server. The server extensions are currently only available to QuickPlace, but in time this technology could be available to Domino designers.

The QuickPlace model

QuickPlace and Notes share the same underlying structure, which is the storage of data as notes in databases. A QuickPlace is a collection of databases -- a database to represent the top-level room of a QuickPlace, a members directory that manages data pertaining to members and their access rights, a search database, and page library databases that manage the documents for each inner room in a QuickPlace.

Objects in a QuickPlace map to objects in a Domino/Notes application. For example, a page in a QuickPlace is the same as a note plus a form and subform in a Domino/Notes application.

The following table maps the objects in a QuickPlace to objects in a Domino/Notes application.

QuickPlace object	Domino equivalent	Object description
Place	File directory	The QuickPlace site, organizes pages in rooms and folders.
Room	Database (.nsf)	A collection of pages with discrete security and authentication.
Folder	Folder/view	An organizing structure for collecting and displaying related pages in a site.
Page	Note + form + subform	The basic vehicle for content. You can create content using the QuickPlace editor or import content from an external source.
Member	Note in contacts1.nsf database	A member note contains information about a team member of a QuickPlace. In addition to this data, the member must be listed in the ACL of main.nsf and names.nsf to pass authentication.
Form	A data note comparable to a Notes form	Manages the display of data notes. A form can contain fields for containing data and use scripts to process and compute data.
Field	A data note comparable to a Notes field	Allows for user input of data into data notes.
Place type	Database template (.ntf) or database (.nsf)	The structure and design for the top level room of the resulting place.
Room type	Database template	The structure and design for the

	(.ntf) or database (.nsf)	resulting room.

Customizing a QuickPlace from the Notes Client

Because QuickPlace shares common data structures with Notes applications, you can customize a QuickPlace from a Notes client by directly manipulating the properties of QuickPlace objects. You can do this using the QDK.

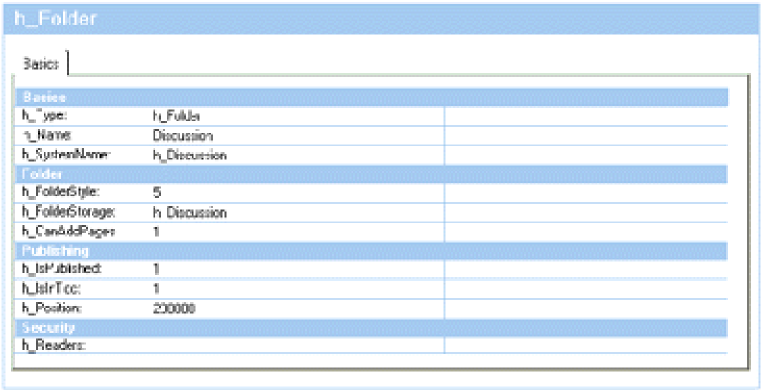
The QDK includes forms you can add to QuickPlace databases to inspect and customize objects, agents for automating the customization, a sample DLL (Dynamic Link Library) for trapping and reacting to QuickPlace commands, and documentation describing how to use these tools.

The following examples describe things you can do with the Notes client and the QDK.

Modifying QuickPlace forms

To customize a QuickPlace from a Notes client, you need to be able to inspect the QuickPlace objects and their associated properties. If you create a QuickPlace using the template files that ship with the QDK, you can inspect each object in your QuickPlace using the forms supplied with the templates. You can then change a field value on a form to modify the QuickPlace object.

The following form in the QuickPlace database main.nsf displays the fields that define a folder called Discussion. Changes you make to field values on this form update the folder in your QuickPlace. You might change the name of the folder to Marketing Discussion or add a list of authorized users to the Readers field to restrict access to this folder. When you save the changes, you can then see the changes in the QuickPlace.



h_Folder

Basics	
h_Type:	h_Folder
h_Name:	Discussion
h_SystemName:	h_Discussion
Folder	
h_FolderStyle:	5
h_FolderStorage:	h_Discussion
h_CanAddPages:	1
Publishing	
h_IsPublished:	1
h_IsInToc:	1
h_Position:	200000
Security	
h_Readers:	

Automating QuickPlace operations with LotusScript agents

Another way to add value to a QuickPlace application is to use agents from the Notes client to automate operations in a QuickPlace. For example, you can use an agent to automate the process of adding new users to a QuickPlace. When you add a new user, you create a new document, assign a username and password and set access rights. The QDK includes the "Create Members from a File" agent that automates this process so you can add batches of new users to a QuickPlace.

The agent reads the text file containing a list of users, passwords, and access levels, and writes data notes to the contacts1.nsf database -- the QuickPlace database that contains most of the user information. It then adjusts access control lists where appropriate and updates the h_Members group document for the QuickPlace you are customizing. You can either run the agent "as is" or use parts of this code to create an agent that suits your needs.

Here are step-by-step instructions on using the "Create Members from a File" agent:

1. Create a text file listing each member's name, password, and access level (Reader, Author, or Manager). If the name contains a space, it must be enclosed in quotes. Use commas to separate the elements. For example:
   ```
   "joe manager", pw, manager
   joeauthor,pw,author
   ```
2. Save the text file in your data directory.
3. To run the agent, copy the database containing the agent to the server containing the QuickPlace you are customizing.
4. Choose Actions - Create Members from a File.
   The agent prompts you to supply a path to the QuickPlace, the name of the QuickPlace, the input file name and the server name.
   The agent creates a member document for each name in the list and assigns the appropriate access level.

Here is the code behind the "Create Members from a File" agent:

```
Sub Initialize
    'Reads user data from file in the following format:  username,
    password, [reader|author|manager]
    'User must enter path to quickplace, quickplace name, data file and
    server when agent is run

    DirName$ = Inputbox("Enter path to quickplace (e.g., c:\quickplace)")
    QPName$ = Inputbox("Enter quickplace name")
    DataFile$ = Inputbox("Enter input file name (must be in data
    directory)")
    Server$ = Inputbox("Enter server name")

    Dim ContactsDb As New notesdatabase ( "", DirName$ +
    "\data\quickplace\" + QPName$ + "\contacts1.nsf")
    Dim MainDb As New notesdatabase ( "", DirName$ +
    "\data\quickplace\" + QPName$ + "\main.nsf")
    Dim NamesDb As New notesdatabase ("", DirName$ +
    "\data\names.nsf")

    Dim MainACL As notesacl, ContactsACL As notesacl
    Dim Entry As notesaclentry
    Dim View As notesview
    Dim MemDoc As notesdocument, Doc As notesdocument
    Dim Item As notesitem

0   'data file
    FileNum% = Freefile()
    FileName$ = DirName$ + "\data\" + DataFile$

    RestOfSysName$ = "/OU=" + QPName$ + "/OU=QP/O=" + Server$
    RestOfACLName$ = "/" + QPName$ +"/QP/" + Server$

    'get handle on h_Members group in the NAB
    Set View = NamesDb.getview("Groups")
    Set MemDoc = View.getdocumentbykey("h_Members/" + QPName$)
    Set Item = MemDoc.getfirstitem("Members")

    Open FileName$ For Input As FileNum%

    Do Until Eof(FileNum%)
        Input #FileNum%, UserName$, Password$, Level$
```

```
0       Set Doc = New notesdocument (ContactsDb)
        '
        'assign field values
        Doc.h_Type = "h_Member"
        Doc.h_Name = UserName$
        Doc.h_SystemName = "CN=" + UserName$ + RestOfSysName$
        Doc.h_Password = Password$
        Doc.h_IsPublished = "1"
        Doc.h_IsHidden = "1"
        Doc.h_SetReadScene = "h_MemberRead"
        Doc.h_SetEditScene = "h_MemberEdit"
        '
        'save so that h_Password has a value
        Call Doc.save (True, True)
        '
        'hash password and resave
        NotesMacro$ = "@Password(h_Password)"
        Doc.h_Password =  Evaluate(NotesMacro$, Doc)
        Call Doc.save (True, True)
        '
        'make users reader, author or manager and set ACLs
        appropriately in main.nsf and contacts1.nsf
        Set MainACL = MainDb.acl
        Select Case Level$
        Case "reader":
            Set Entry = MainACL.createaclentry (UserName$ +
            RestOfACLName$,  ACLLEVEL_READER)
        Case "author":
            Set Entry = MainACL.createaclentry (UserName$ +
            RestOfACLName$,  ACLLEVEL_AUTHOR)
            Entry.ispublicwriter = True
        Case "manager":
            Set Entry = MainACL.createaclentry (UserName$ +
            RestOfACLName$,  ACLLEVEL_MANAGER)
            Entry.candeletedocuments = True
            '
            'managers must be added to contacts1.nsf ACL as well
            Set ContactsACL = ContactsDb.acl
            Set Entry = ContactsACL.createaclentry (UserName$ +
            RestOfACLName$,  ACLLEVEL_MANAGER)
            Entry.candeletedocuments = True
            Call ContactsACL.save
        End Select
        Call MainACL.save
        '
        'all users must also be added to the h_Members group in
        names.nsf
        Call Item.appendtotextlist("CN=" + UserName$ +
        RestOfSysName$)
        Call MemDoc.save (True, True)
    Loop
    Close FileNum%
End Sub
```

Automating QuickPlace operations using the Event Interface
Another way for Designers to customize a QuickPlace is to automate operations using the QuickPlace Event Interface included with the QDK. This allows you to write programs that trap QuickPlace commands and execute routines. For example, a sample program that ships with the QDK uses a C++ routine to send customized e-mail invitations to new members of a QuickPlace.

The code and a full description of how this sample application works is available as part of the QDK release 1.0.2 download.

The future of QuickPlace

An upcoming release of QuickPlace will further strengthen the position of this product not only as an out-of-the-box solution for collaborative applications, but as a customizable platform for the development and delivery of custom collaborative applications. With its roots in the Domino/Notes family, QuickPlace will remain a product to which Domino/Notes application developers can add significant value for their corporate customers.

ABOUT THE AUTHOR

Julie Forgo, a technical writer with more than 12 years' experience, is the project leader for the Domino Designer *Application Developer's Guide* and recently worked on the QuickPlace Developer's Toolkit. When not busy extracting feature details from developers, Julie chases after her three children and performs miracles with a hot-glue gun.