

Ten Ways SharePoint 2010 Will Impact Your Notes Migration

Written by
Steve Walch
Senior Product Manager, Quest Software, Inc.

© 2010 Quest Software, Inc.
ALL RIGHTS RESERVED.

This document contains proprietary information protected by copyright. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose without the written permission of Quest Software, Inc. ("Quest").

The information in this document is provided in connection with Quest products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest products. EXCEPT AS SET FORTH IN QUEST'S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software World Headquarters
LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656
www.quest.com
E-mail: legal@quest.com

Refer to our Web site for regional and international office information.

Trademarks

Quest, Quest Software, the Quest Software logo, AccessManager, ActiveRoles, Aelita, Akonix, AppAssure, Benchmark Factory, Big Brother, BridgeAccess, BridgeAutoEscalate, BridgeSearch, BridgeTrak, BusinessInsight, ChangeAuditor, ChangeManager, Defender, DeployDirector, Desktop Authority, DirectoryAnalyzer, DirectoryTroubleshooter, DS Analyzer, DS Expert, Foglight, GPOAdmin, Help Desk Authority, Imceda, IntelliProfile, InTrust, Invirtus, iToken, IWatch, JClass, Jint, JProbe, LeccoTech, LiteSpeed, LiveReorg, LogAdmin, MessageStats, Monosphere, MultSess, NBSpool, NetBase, NetControl, Npulse, NetPro, PassGo, PerformaSure, Point,Click,Done!, PowerGUI, Quest Central, Quest vToolkit, Quest vWorkSpace, ReportAdmin, RestoreAdmin, ScriptLogic, Security Lifecycle Map, SelfServiceAdmin, SharePlex, Sitraka, SmartAlarm, Spotlight, SQL Navigator, SQL Watch, SQLab, Stat, StealthCollect, Storage Horizon, Tag and Follow, Toad, T.O.A.D., Toad World, vAutomator, vControl, vConverter, vFoglight, vOptimizer, vRanger, Vintela, Virtual DBA, VizionCore, Vizioncore vAutomation Suite, Vizioncore vBackup, Vizioncore vEssentials, Vizioncore vMigrator, Vizioncore vReplicator, WebDefender, Webthority, Xaffire, and XRT are trademarks and registered trademarks of Quest Software, Inc in the United States of America and other countries. Other trademarks and registered trademarks used in this guide are property of their respective owners.

Contents

- Introduction 3
- Ten Ways SharePoint 2010 Will Impact Your Notes Migration 4
 - 1. Scalability 4
 - 2. Managed Metadata 4
 - 3. Office Integration 5
 - 4. Offline Capabilities 6
 - 5. SharePoint Online 7
 - 6. Improved Content Pages 8
 - 7. Code-Free Development Capabilities 9
 - 8. InfoPath List Forms 9
 - 9. Declarative Workflow 10
 - 10. Business Connectivity Services and External Lists 11
- Conclusion 13
- About the Author 14

Introduction

Over the past five years, many organizations have decided to abandon their legacy Lotus Notes/Domino environments and transition to a platform based on Microsoft collaboration technologies (Exchange Server, SharePoint and Office). Many of these organizations, though, have hesitated in tackling the most difficult part of this transition: migrating their applications from Notes to SharePoint. Some hesitate due to concerns about the cost of rebuilding their applications on SharePoint, while others worry whether SharePoint has all the capabilities that their applications would need.

The release of Microsoft SharePoint Server 2010 and SharePoint Foundation 2010 has changed the situation dramatically, for two reasons. First, Microsoft added a great deal of functionality that directly targets the needs of enterprises migrating from Lotus Notes. Second, Microsoft significantly improved the tools for developing custom applications in SharePoint, thus reducing the time and cost required to rebuild applications.

This paper discusses 10 aspects of SharePoint 2010 that will change the game for enterprises of all sizes who want to use SharePoint to replace or enhance their Notes environments. The first six items are major platform and functionality improvements that Notes people should especially appreciate and will change their thinking about how applications might be rebuilt. The last four items address improvements of particular interest to developers who want to rebuild the functionality of their Notes applications on SharePoint.

Ten Ways SharePoint 2010 Will Impact Your Notes Migration

1. Scalability

It is not unusual for Notes databases in large enterprises to contain tens of thousands of documents. Organizations attempting to move this content to SharePoint 2007 ran into some pretty severe size limitations on SharePoint lists and libraries. The most painful one was that only 2,000 documents could be displayed in any one view or folder; if you had 10,000 documents, you had to split them into multiple folders or find another way to balance the load. These workarounds not only slowed down the migration process, but sometimes resulted in a structure that was unusable. Ignoring the limits, on the other hand, as many did, meant risking poor SharePoint performance.

With SharePoint 2010, however, the recommended maximums for many items have more than doubled. Even better, the penalty for exceeding the limits is far less severe: performance does not drop off dramatically once you exceed the recommended limits, and SharePoint will now automatically throttle certain operations that previously would have brought your servers to their knees.

A more subtle aspect of scalability is the improvements in SharePoint 2010 search technologies. It is now a lot easier to build great search solutions for your users without putting everything in one huge view. This makes it feasible to start separating your content management decisions (how to organize your sites, lists and libraries) from your user interface decisions (how users will navigate the results).

2. Managed Metadata

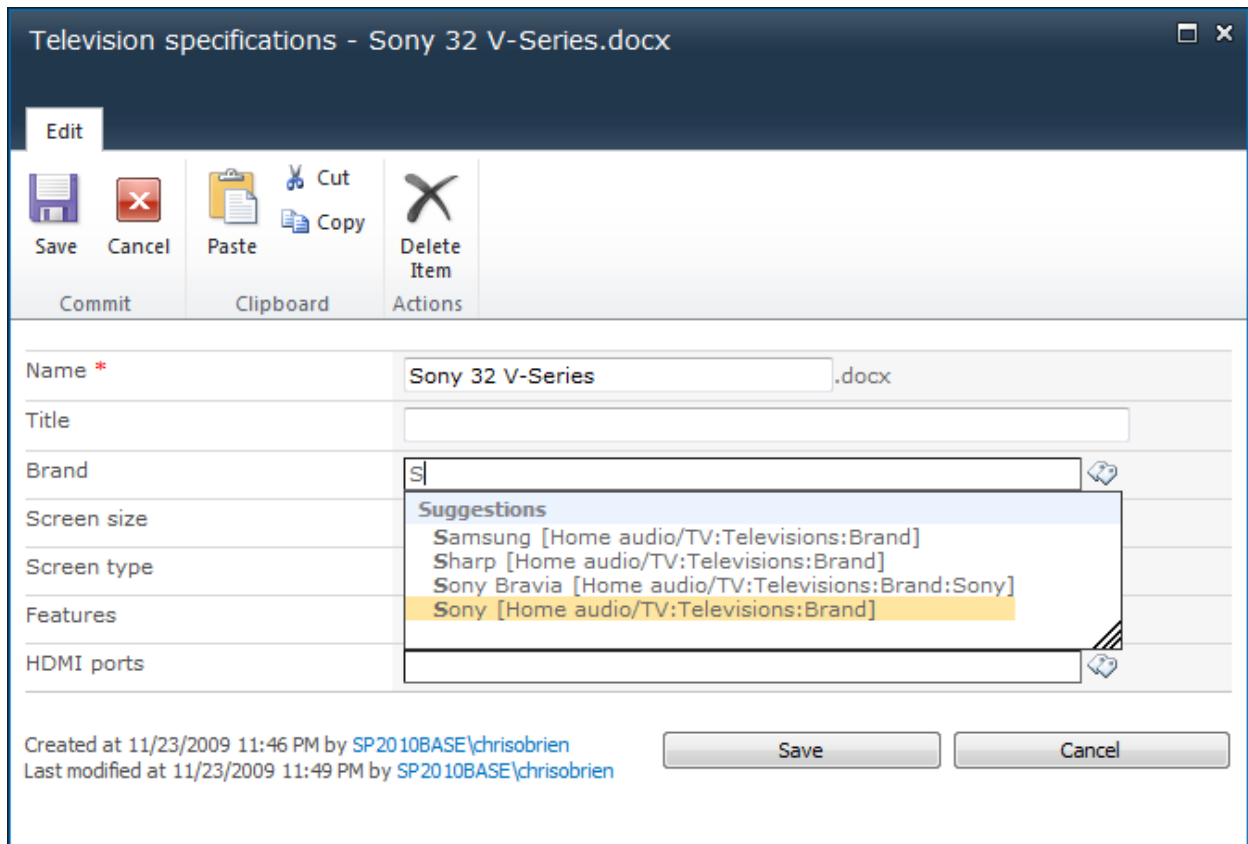
Closely tied to the new search capabilities discussed above is Managed Metadata. Central to most custom Notes applications are keyword fields, which range from simple pop-up or drop-down lists of terms that users can select from (for example, product codes) to lookups in other views and databases (for example, customer numbers). Some Notes solutions even include custom term-management databases where users could centrally manage the terms, codes, and other resource documents used by other Notes applications.

SharePoint 2007 provided two pretty good ways to achieve this functionality: *Choice fields* allowed users to select from a simple set of choices, and *Lookup fields* allowed them to select records from a different list within the same site. These options provided much of the required functionality, but in some situations, these features were too simplistic and their scope was too limited.

A powerful new feature of SharePoint 2010 is *Managed Metadata* (also referred to as Enterprise Metadata). You can think of this as the next level beyond Choice fields and Lookup fields. Now SharePoint enables you to manage all the terms that are important to a particular application domain (the keywords, product codes, customer types, document categories, etc.) in a shared *term store*. SharePoint 2010 includes a complete interface that enables administrators to maintain terms, along with the following cool features:

- **Aliases** – Different words that mean the same thing
- **Translation** – Different words in different languages that mean the same thing
- **Context** – The same word can mean different things in different settings
- **Hierarchy** – Organization of terms into categories and sub-categories

Some people think of the term store as a corporate taxonomy. But while the term store does attempt to address the needs of a corporate taxonomy, it conjures up images of full-time taxonomy managers and expensive knowledge management consultants trying to establish the perfect set of terms for the enterprise. This may be well beyond the scope of many SharePoint teams. Happily, the Managed Metadata feature in SharePoint 2010 scales very well from simple needs to enterprise knowledge management solutions. For example, you can easily create a simple Managed Metadata field and a term set that is scoped to just one SharePoint site; multiple lists and libraries within that site can reference that same term set just as easily. Later, if you need to, you can scale out to enterprise term sets that span many site collections and can even be replicated between multiple SharePoint farms.



Managed Keywords offer a new way to migrate Notes keyword fields that pull data from centralized term stores.

When you use a term store to classify (or, to use the modern buzzword, “tag”) your content, you get additional benefits. Not only are you using a consistent set of terms across multiple applications, but you can use these terms to find content more easily. For one thing, SharePoint search is tuned to work with managed metadata to search across many applications that use the same term sets.

Of special interest to Notes developers is the ability to quickly build drill-down views of your content based on your metadata. This can replace the proliferation of categorized views within many legacy Notes applications that allowed users to navigate their content in various ways. You can also filter views based on your metadata so you can eliminate many views that simply showed different subsets of your data.

3. Office Integration

For the last 15 years, Notes developers and third-party product companies around the world have been trying to achieve good integration with Microsoft Office. The goals in this area include:

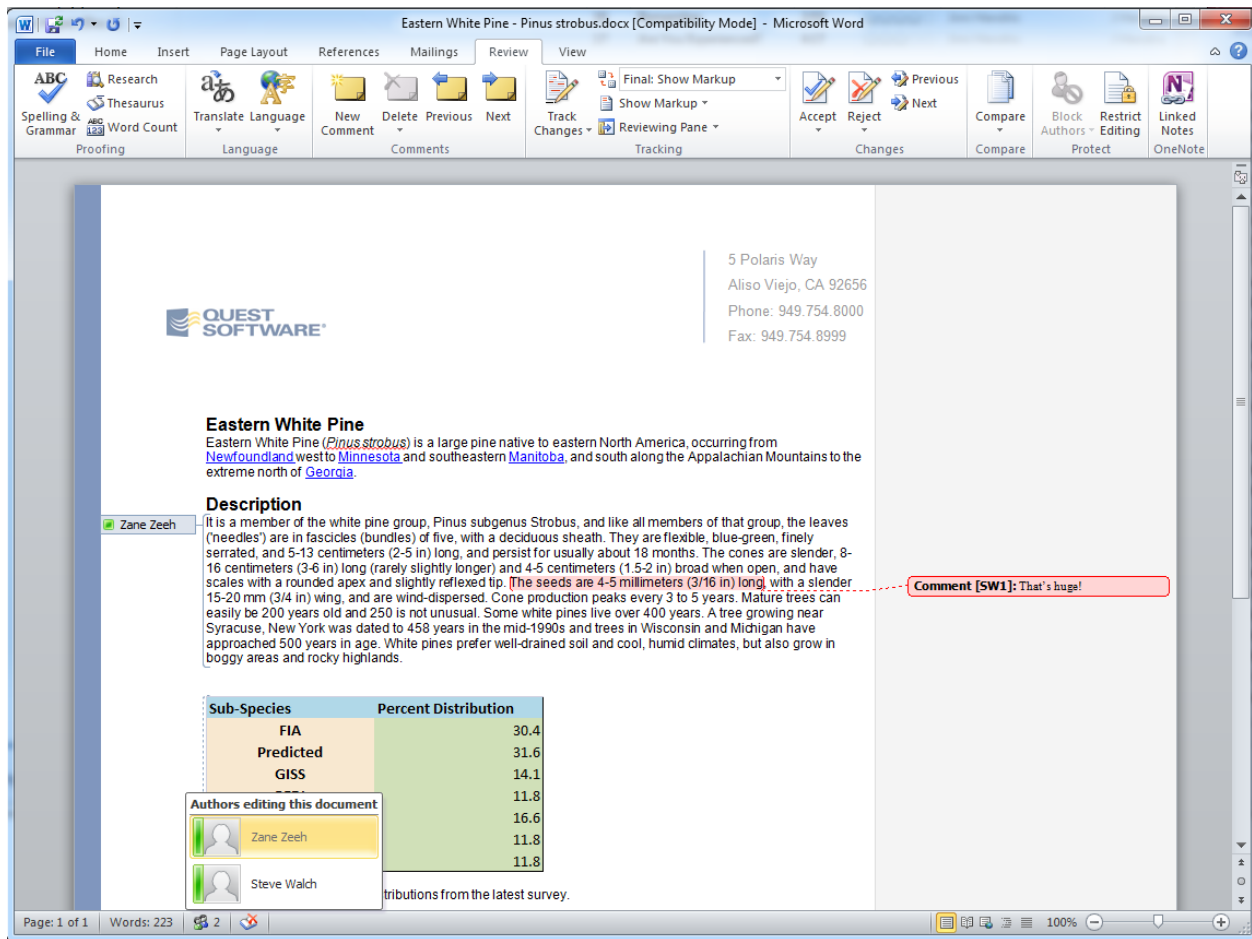
- Creating a nice experience for end users who want to edit an Office document attached inside a Notes document
- Externalizing metadata of an Office document so it could be displayed in a Notes view
- Generating an Office document using data captured in Notes

These solutions were often expensive to develop, and the results were often unsatisfying, clunky, and fragile.

However, as part of the Office family, SharePoint has always had excellent Office integration. For example, the integration between Office client applications and SharePoint document libraries enable users to open Office documents from libraries, edit them, and seamlessly save them back again. If a version control, check-in/check-out, or approval process or workflow has been enabled for the document library, it will work. Office documents in SharePoint are easy to search, and you can even generate an instant SharePoint workspace to allow teams to collaborate on a particular document.

Because of these capabilities, Notes applications that have traditionally been containers for Office documents transition particularly well to SharePoint. In addition, many organizations elect to *convert* certain types of Notes

documents to an Office format. In particular, the ability to reformat rich text Notes documents into great-looking Word or InfoPath documents has been a popular feature for high-end migration tools.



When you migrate your Notes documents to a Microsoft Word format, your users can leverage new 2010 features such as collaborative editing of a single document.

With SharePoint 2010, things are getting even better. On the top of almost everyone's list of desired SharePoint 2010 features is collaborative editing of Office documents. For instance, multiple users can open the same Word document and edit it at the same time. Users can see the changes being made by other users almost instantly thanks to a sophisticated change-tracking interface. Behind the scenes, SharePoint is coordinating this interaction between the various users as well as managing the revisions.

Another impressive new feature is the new Office Web Apps. These are completely browser-based versions of Word, Excel, PowerPoint, and other applications that allow users to view and edit documents in SharePoint without having to install Office clients. While these web editors do not have *all* the capabilities of their fat-client counterparts, they do cover the needs of most users and allow for many new compelling use cases.

4. Offline Capabilities

Even though we now count on nearly ubiquitous Internet connectivity and bandwidth, many legacy Notes applications depend on the ability to go offline. Notes has always been famous for its ability to replicate whatever data you needed to your laptop and (assuming your application was designed to work that way) enable you to continue working while unplugged, such as on the airplane, at the customer site or at the vacation home where you can't access the Internet.

SharePoint was designed for a modern, connected world and by itself, does not accommodate offline use well. With SharePoint Workspace 2010, however, you can synchronize lists, libraries, and entire sites to your laptop, use them offline, and then sync up with the site the next time you are connected. This includes not only the data, but much of the logic of the site (including custom data entry forms), resource data (such as Lookup lists that your application depends on), and more. When you click the "Sync to My Computer" button, you can choose to download everything (the default), just the application but not the content, or a customized selection of available content.

SharePoint Workspace includes a lot of smarts about how application functionality should work when users go offline. For example, certain workflow operations will execute only when new documents are posted back to the server. Also, the offline application will try to connect to an online resource that the user decided not to replicate locally (so offline does not have to be all or nothing).

Not surprisingly, much of this functionality is based on Groove technology, which was built by many of the same developers who worked on the original Notes product. The first few releases of Groove were not all that compelling for many Notes customers because it required too much special setup, development and user training. But Microsoft has done a great job of taking the best of the Groove plumbing and repackaging it as a very compelling offering that fulfills most of the “offline” needs of a typical Notes application.

Perhaps the best part of this new technology is the way SharePoint Workspace works with the new application development capabilities that have been added to SharePoint, SharePoint Designer, InfoPath and other Office applications. Even the External Lists feature (described below in “10. Business Connectivity Services and External Lists”) can be made to work offline.

5. SharePoint Online

Most organizations considering a move from Notes to the Microsoft platform have also considered having their environment hosted. The Microsoft Business Productivity Online Suite (BPOS), also referred to as Microsoft Online (MSO), is an obvious choice. BPOS includes Microsoft Exchange Online, Microsoft SharePoint Online, Microsoft Office Live Meeting, and Microsoft Office Communications Online. There are two versions of BPOS:

- **BPOS-D (Dedicated)** – Large enterprises have dedicated private hosted servers
- **BPOS-S (Standard)** – Multiple customers share one hosted environment.

While Exchange Online has been a very popular choice for replacing Notes mail/calendar environments, movement to SharePoint Online has been much slower, for two reasons:

- The capabilities for customizing SharePoint Online Standard were very limited because there was no way for the hosting service to ensure that customized code did not cause problems. SharePoint Online Dedicated is not quite as restricted, but customizations do involve a time-consuming approval process.
- SharePoint Online Standard did not provide any means for migrating legacy content to the hosted servers. The only interfaces for writing content were quite limited; for example, a migration tool would be unable to preserve the author names and created/modified dates of Notes documents. SharePoint Online Dedicated gave you a few additional options, but none of them were as simple as migrating to on-premises environments.

SharePoint 2010 addresses these issues with new capabilities and interfaces for external tools. In fact, SharePoint 2010 offers many improvements specifically designed to make life easier for hosting companies and their customers, particularly in the areas of customization and development.

To reduce the limits on customization, SharePoint 2010 offers a “sandbox,” which provides a safe way for the server to safely run code written and uploaded by other people. The sandbox environment carefully controls what operations are allowed, ensures that programming errors cannot impact other processes, and throttles the use of system resources to agreed-upon levels. As a result, customers can upload their custom web parts and other SharePoint solutions and they will run just fine, provided they have stayed within the limits of the sandbox.

Many other types of customizations, some of which are described below, are now available in the updated releases of SharePoint Online. Of particular importance is the ability to use SharePoint Designer to directly customize your hosted SharePoint sites, which is important for a wide variety of customizations ranging from changing site styles to designing workflows.

More important, SharePoint Online now includes a set of web services that is adequate for doing serious content migration. Now a migration tool can directly provision sites and lists, migrate schema and design, migrate content with high fidelity, map user names, preserve document links, set permissions, and do all the other things necessary for a quality application migration.

It is worth mentioning that some customers take a hybrid approach: they migrate certain Notes applications to on-premises SharePoint environments and others to SharePoint Online. Organizations may choose to keep some applications in-house due to legal requirements, bandwidth issues, the need for a high level of customization or to connect to line-of-business applications. The good news is that high-end migration tools now give you a choice and allow you to re-target a particular Notes application at any time.

6. Improved Content Pages

Content in the SharePoint world exists not only in list items and Office files in document libraries; we also have pages. The idea of ASPX pages living in a document library can be confusing at first. Aren't ASPX pages things that developers think about when building web applications? Yes, they are, but SharePoint offers a clever way to use ASPX pages as content pages.

The key to this capability is to fully separate the parts of the page that the developer has to think about (the controls, the layout and any associated code) from the parts that the author wants to think about (the content). The result is that users can create a page just as they would create any other kind of document.

Unlike list items, which are generally form-oriented with lots of fields, content pages tend to be one big section of rich text. It really is (or appears to be) just a web page where the authors can add any content they want. SharePoint 2007 offered a few different kinds of content pages: basic pages, wiki pages, web part pages, and publishing pages. This provided a nice range of choices for site creators and application developers to use when designing SharePoint sites. It was also possible to migrate rich text Notes documents to these various page types using a high-end migration tool. Unfortunately, the capabilities of these pages were somewhat limited and the editing experience for end users was mediocre.

The screenshot shows a SharePoint Wiki page for 'Eastern White Pine'. The page includes a navigation breadcrumb (Home > Notes Migration - Wiki Library > Eastern White Pine - Pinus strobus), a search bar, and a left-hand navigation pane with sections for Libraries, Lists, and Discussions. The main content area features a title, a description, a 'Description' section, and a table showing the distribution of sub-species.

Sub-Species	Percent Distribution
FIA	30.4
Predicted	31.6
GISS	14.1
GFDL	11.8
Hadley	16.6
UKMO	11.8
CCC	11.8

The above table shows the distributions from the latest survey.

Wiki pages provide an excellent alternative format for migrating certain types of Notes applications.

In SharePoint 2010, content pages have been greatly improved. In particular, wiki pages are now a very useful way to store rich content. Microsoft basically merged the concept of basic pages, wiki pages and web part pages into a single construct that is much more powerful and easier to use. Now the rich text editor sizzles. End users can upload images and files and even embed web parts in their rich content (similar to the way they used to embed OLE objects in their Windows applications).

Wiki pages are now fairly ubiquitous in standard SharePoint templates. When you create a new "page" using the Create menu in a team site, you are now creating a wiki page. And the Site Pages library that (by default) holds pages that users create is now the Wiki Page Library.

If you have a Notes application that was intended to manage a bunch of rich text pages, consider migrating them to wiki pages in SharePoint 2010. QuickPlace and Quickr sites are often good candidates for migration to wiki pages. However, if you want to utilize the approval workflows and other features of some of the high-end publishing templates in SharePoint, then publishing pages are still a good choice.

7. Code-Free Development Capabilities

There are good tools available today for migrating the content of your Notes applications to SharePoint, but what about the design? One of the biggest barriers to moving large numbers of Notes applications to SharePoint is the cost of rebuilding complex applications.

Some organizations assume that rebuilding a Notes application in SharePoint will require the same number of hours and lines of code that was needed to create it in Notes. This is far from the truth. The reality is much better: SharePoint allows you to do many things out of the box or via simple configuration that Notes developers had to build from scratch.

For example, in SharePoint 2007, version control can be enabled in any list or library in seconds. Similarly you can enable check-in/check-out functionality in any document library. Simple approval processes (pending, approved, rejected) are also instantly available and simple review processes are available as canned workflows. The Office integration and various site templates (publishing solutions, team rooms, meeting workspaces) are further examples. All of these are immediately recognizable by someone in the Notes world as being in the realm of *custom development projects*.

SharePoint 2010 offers even more code-free development capabilities (in addition to all the new “big ticket” items, such as the Managed Metadata feature described above). One that will be particularly popular with Notes developers is the ability to specify simple input validation formulas for individual fields or for the document as a whole, including the messages that should be displayed if the tests fail. You can also add multi-level column indexes, specify which columns require unique values, and even enforce “relational” constraints between lists. Displaying content per-user or per-location is easier, and the new toolbar makes customizing pages and views easier as well.

Moreover, SharePoint Designer enables you to perform deeper design of your sites, without writing code. For instance, you can design your declarative workflows and define external lists (both of which are described below). You can also perform more extensive page customizations, edit master pages, work with complex web parts, add custom action buttons, and much more. SharePoint Designer is now a free download for any developer or power user who wants to customize sites beyond what can be done in the browser, but still without writing code.

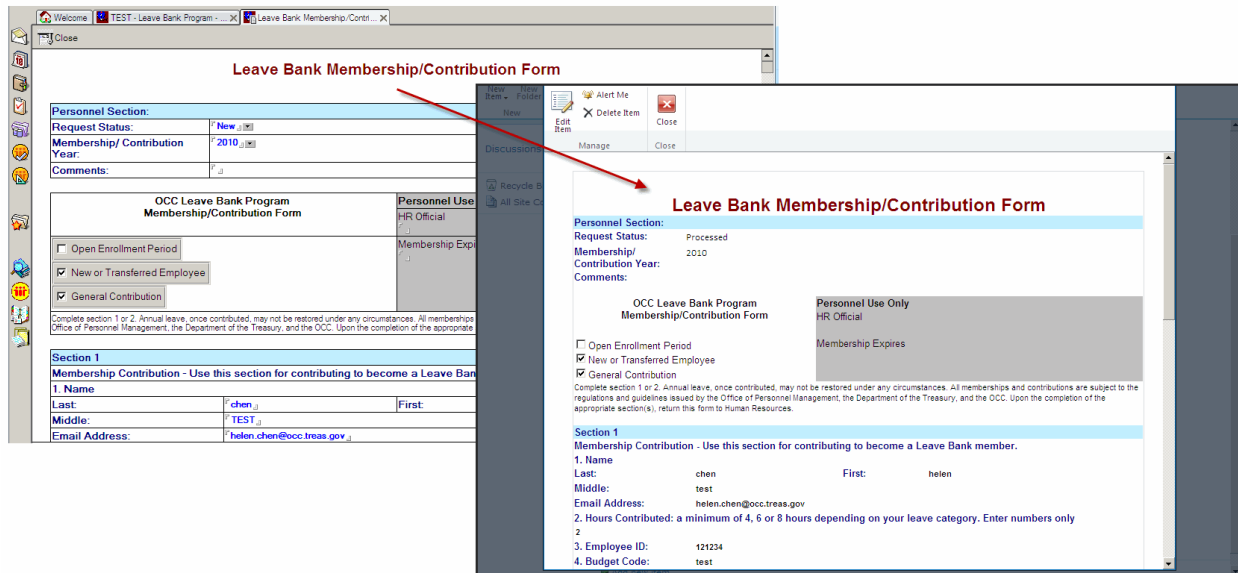
One great way to expand your code-free development options even further is to use third-party web parts. Some web part products on the market today have been designed to help specifically with reproducing Notes-like functionality in SharePoint at a much lower cost than developing it yourself.

In short, the list of things you can accomplish without resorting to Visual Studio has been significantly expanded. Of course, there is, and probably always will be, plenty of cases where “serious coding” is needed, but by keeping these cases to a minimum. SharePoint 2010 can reduce your migration costs.

8. InfoPath List Forms

One of the most important aspects of migrating a Notes application to SharePoint is form design. InfoPath is, of course, Microsoft’s primary solution for building data entry forms for complex business documents. Microsoft InfoPath 2010 is a forms-creation and data-gathering tool that helps organizations streamline business processes. InfoPath is designed for both advanced business users and developers, depending on the type of forms-based solution an organization needs. Many Notes customers see InfoPath as the best choice for migrating Notes forms that contain user-friendly field layouts, non-trivial data validation rules, and interactive functionality such as hide-when formulas. There are great tools available for migrating Notes form designs to InfoPath templates and for migrating Notes documents to InfoPath data documents as well.

With SharePoint 2007, the decision to use InfoPath brought some unfortunate limitations. First, you were forced to store your data as XML data documents, usually in SharePoint document libraries. InfoPath forms did not really work with SharePoint lists—the natural target for most Notes application content. Unless you were willing to deploy the InfoPath 2007 client to all your user’s desktops, you had to live with the limitations of InfoPath Form Services. (Try telling your Notes users that they can’t have embedded images anymore.) Finally, the process of building, maintaining and publishing InfoPath 2007 forms was a cumbersome and often fragile process that was pretty much outside of “normal” SharePoint customization and development activities.



Migrating a Notes form to an InfoPath List form helps preserve the look and feel of your Notes application without a lot of development effort.

SharePoint 2010 introduces a great new way to use InfoPath: InfoPath List Forms. This feature allows you to use InfoPath forms as your editor for list items. Now you get the best of both worlds: a lightweight way to store documents with custom schema and a great way to design custom forms for entering and displaying them. All this, by the way, closely mirrors how things were structured in Notes.

In addition, the experience for developers is now greatly simplified (and far better than it ever was in Notes). Even junior developers will be able to design custom forms using an integrated set of tools. They can even start with an auto-generated InfoPath form (based on your list schema) and start customizing it from there.

InfoPath List Forms rely on InfoPath Form Services, which means that everything runs in the browser. Happily, there have been a number of dramatic improvements in InfoPath Form Services, and the web experience really is at near-parity with the rich client experience. And there are plenty of other InfoPath improvements that will reduce the cost of migrating Notes applications, including additional controls, more seamless workflow integration, and automatic offline support.

9. Declarative Workflow

When people think of complex custom Notes applications, they think of workflow. Unfortunately, although migration tools can do a great job migrating schema, content, security and forms, they often have a hard time translating Notes workflows to SharePoint workflows.

Notes workflows are almost always implemented as code attached to various buttons, form events, and agents. By contrast, the Microsoft platform encourages the use of declarative workflows—the workflow is expressed as a set of rules that can be entered, modified, and (best of all) understood by a non-programmer. You can write code if you need to, but this code is usually confined to “activities,” the units of action that you wire together in your declarative workflow.

Declarative workflow capabilities have improved significantly in SharePoint 2010. First of all, you get a much richer set of out-of-the-box workflows to use in your applications. This list depends on what edition of SharePoint you install and which SharePoint template you are using, but at a minimum it should include an Approval Process, a Collect Feedback review process, a Collect Signatures review process, and a Disposition Approval workflow for managing content lifecycles. You can now customize the out-of-the-box workflows if needed or design your own from scratch.

If you do need to design your own workflows, you can benefit from the new capabilities in SharePoint Designer. This option was somewhat limited in SharePoint 2007, which forced many companies to use Visual Studio instead, but those days are gone. Now SharePoint Designer workflows can address the need of the majority of the workflows found in Notes applications without having to write code.

For example, users have a much bigger vocabulary of workflow “conditions” and “actions” to choose from, including Notes-like options such as changing the permissions on a document after it is submitted for approval, sending mail notifications, looking up a person’s manager, and moving content to another location. A particularly powerful feature

enables you to impersonate other users during a workflow to perform certain actions using the workflow author's credentials. You can even use InfoPath to design nicer forms for collecting information from end users.

You can now design workflows that operate at the site level or the list level, and you can reuse workflows across an entire site collection and even export them to use on a completely different SharePoint farm. Workflows can be visualized as a set of human-readable steps and can even be exported to Visio for graphical display.

The takeaway here is twofold. First, you can replace most of your Notes workflows without writing code; save your expensive developers for the hard stuff. Second, do not attempt to translate all the code that your Notes developers wrote into code on the Microsoft platform. Instead, devote your energies to *understanding* what your old workflows did and designing how to accomplish these same goals using declarative workflows.

10. Business Connectivity Services and External Lists

Notes had a lot of capabilities for connecting to external systems like relational databases and enterprise applications such as SAP and PeopleSoft. As a result, some Notes applications that will be replaced don't really have much content of their own—they are just front ends for some back-end system.

SharePoint 2007 has a somewhat limited facility for connecting to external data: the Business Data Catalog. In SharePoint 2010, this is replaced by a much more powerful feature called Business Connectivity Services (BCS). You can use this new facility to search, read, and write content in your back-end systems. BCS introduces External Content Types, which are basically definitions of where your external data lives and how to read and write entities in your back-end data source.

This feature gets really powerful when you expose this content as an External List. An External List looks and feels like a native SharePoint list, and you can easily include it as part of any SharePoint page. You can design views, display the content in custom web parts, and even build custom InfoPath forms for displaying and editing data. It really is that simple.

SharePoint gives you a rich set of choices for adding custom actions, integrating security and even implementing single sign-on. Some of these more advanced features involve a modest learning curve, but the good news is that you can get basic read/write functionality working very quickly, and you can do a lot without having to be a programmer. You can use the powerful facility without writing any code if your external data is in SQL Server or available via web services. For more complex situations, you might have to write some code to build a .NET class that connects to your external data source.

The screenshot displays a SharePoint External List view for 'QDemo External List'. The main content area shows a table with the following data:

Title	CreatedBy	CreatedDate	DocID
White Oak - Quercus Alba	Kimberly Piper/QDemo	11/2/2007 2:13 AM	2629
Bur Oak - Quercus macrocarpa	Lynda Vincent/QDemo	11/2/2007 2:39 AM	2630
Swamp White Oak - Quercus bicolor	Sonya Nye/QDemo	11/2/2007 2:46 AM	2631
Chestnut Oak - Quercus montana	John Lenfestey/QDemo	11/2/2007 2:54 AM	2632
Eastern White Pine - Pinus strobus	John Lenfestey/QDemo	11/2/2007 2:56 AM	2633
Minnesota	Brian Knutson/QDemo	11/2/2007 2:58 AM	2634
Georgia	Brian Knutson/QDemo	11/2/2007 3:00 AM	2635
Newfoundland	Brian Knutson/QDemo	11/2/2007 3:01 AM	2636
Manitoba	Brian Knutson/QDemo	11/2/2007 3:03 AM	2637
Quebec	Beverly Amundson/QDemo	11/2/2007 3:08 AM	2638
Florida	Beverly Amundson/QDemo	11/2/2007 3:11 AM	2639
Texas	Tim Fountain/QDemo	11/2/2007 3:15 AM	2640
Some links	Stephen Walch/Proposition	7/28/2008 4:23 PM	2641
Two body attachments and two doc attachments	Stephen Walch/Proposition	11/4/2009 7:37 PM	2642
Office attachments	Stephen Walch/Proposition	11/18/2009 2:14 AM	2643
PPT Objects	Stephen Walch/Proposition	12/22/2009 11:12 AM	2644
Empty attachment	Stephen Walch/Proposition	12/22/2009 11:17 AM	2645
Blocked files	Stephen Walch/Proposition	5/25/2010 3:15 PM	2646
Attachments with spaces			2647
Two attachments with the same name			2648
White Oak - Quercus Alba			2649
Bur Oak - Quercus macrocarpa			2650
Swamp White Oak - Quercus bicolor			2651

The 'External Data Source Type Selection' dialog box is open, showing a dropdown menu with 'SQL Server' selected. Other options include '.NET Type' and 'WCF Service'.

With External Lists, you can migrate your Notes data to SQL Server and still use SharePoint as the user interface.

For an organization transitioning away from Notes, there are three very interesting possibilities here:

1. You have a powerful tool for replacing the Notes applications that were simply front-ends to external data sources.
2. It is possible to treat your Domino servers as an external data source in SharePoint. With a little effort, you can implement Domino web services that expose a particular Notes database and bring that into SharePoint.
3. You now have choices about where you migrate your Notes data. Instead of migrating your content to SharePoint lists, libraries, and pages, you can choose to migrate certain applications to SQL Server instead. Using External Content Types and External Lists, you can build the user interface of your application in SharePoint. Some migration tools give you a choice about where to put back-end data and also help you migrate your forms and other items to InfoPath and SharePoint. Most people will still want to migrate the bulk of their Notes applications to pure SharePoint solutions, but BCS gives you a realistic alternative when you need one.

Conclusion

At the end of the day, the decision as to what can and should be migrated comes down to cost and capabilities: How much will it cost to migrate your content and application design, and what will your applications look like and be able to do after migration? With the features discussed in this paper, SharePoint 2010 makes migrating from Lotus Notes to SharePoint easier than ever and delivers the functionality organizations need after migration.

About the Author

Steve Walch is a senior product manager at Quest Software. He has been a Lotus Notes and Microsoft technologist since 1993. In 2002, he founded Proposion Software with the mission to connect Lotus Notes with the emerging Microsoft .NET development platform. Over time, Proposion focused on Notes-to-SharePoint migration and integration tools, and became the leading vendor in that market. In 2007, Proposion was acquired by Quest Software. Walch developed Quest Notes Migrator for SharePoint, which has been used by organizations of all sizes around the world to execute successful migrations. His [Notes 2 SharePoint blog](#) is highly regarded by industry experts and analysts.

About Quest Software, Inc.

Now more than ever, organizations need to work smart and improve efficiency. Quest Software creates and supports smart systems management products—helping our customers solve everyday IT challenges faster and easier. Visit www.quest.com for more information.

Contacting Quest Software

PHONE 800.306.9329 (United States and Canada)

If you are located outside North America, you can find your local office information on our Web site.

E-MAIL sales@quest.com

MAIL Quest Software, Inc.
World Headquarters
5 Polaris Way
Aliso Viejo, CA 92656
USA

WEB SITE www.quest.com

Contacting Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a commercial version and have a valid maintenance contract.

Quest Support provides around-the-clock coverage with SupportLink, our Web self-service. Visit SupportLink at <https://support.quest.com>.

SupportLink gives users of Quest Software products the ability to:

- Search Quest's online Knowledgebase
- Download the latest releases, documentation, and patches for Quest products
- Log support cases
- Manage existing support cases

View the Global Support Guide for a detailed explanation of support programs, online services, contact information, and policies and procedures.



5 Polaris Way, Aliso Viejo, CA 92656 | PHONE 800.306.9329 | WEB www.quest.com | E-MAIL sales@quest.com
If you are located outside North America, you can find your local office information on our Web site

© 2010 Quest Software, Inc.
ALL RIGHTS RESERVED.

Quest Software is a registered trademark of Quest Software, Inc. in the U.S.A. and/or other countries. All other trademarks and registered trademarks are property of their respective owners.
WPW-10WaysSPMig-Walch-US-MJ-20100826