



Documents or tables for your Notes data?

LDC Via: a better way to migrate

Overview

Microsoft SharePoint is often presented as the natural destination for IBM Notes content when retiring IBM Domino environments, and there are a number of products in the marketplace aimed at facilitating this move (e.g Quest's "Migrator for Notes to SharePoint").

However, for a document-based system like IBM Domino it surely makes far more sense to move content to another document store than to a relational database like Microsoft SQL Server (the back-end data store for SharePoint).

Considerations when migrating to SharePoint

Planning

Migrating a Domino application to SharePoint, or indeed any system based upon a relational database, means a significant amount of up-front planning. Current migration vendors summarise the steps involved as follows:

- Assessment
- Rationalisation
- Planning
- Design review
- Migration
- Deployment

This is a significant undertaking, making the planning stage especially onerous, and we haven't even begun to assess the Domino application landscape itself yet: which applications *really* need to move? What can be retired? Are there off-the-peg solutions for some of the older bespoke applications?

Design

Assessment and rationalisation result in a candidate list of applications suitable for migration, so next comes the detailed technical planning. This entails mapping individual document attributes to tables and columns in SQL Server (perhaps using form design elements as a starting point).

(There are SharePoint-specific constructs to consider also: will you use lists, libraries, InfoPath forms, or ready-made PDFs?)

Once columns and data types are mapped, these tables will require additional design work: are foreign and primary keys required? What needs indexing? Normalisation? Is there a need to split single Notes document types into multiple related tables? What data types should the tables incorporate?

When it comes to SharePoint, in addition to mapping individual document types (i.e. forms) to SQL Server tables, database attributes come into play. This includes the formation of classification rules, disposition rules, and target site rules, along with options as to whether to migrate Notes users to domain users, and so forth.

There is a lot to consider and a lot to set up.

The Domino specifics

Domino is a document store, one of the original "NoSQL" implementations. So when moving its data to a relational system, how should binary data (i.e. attachments, embedded graphics, rich text) be dealt with? What about Domino-specific constructs like document, database and view links? It's likely you've made extensive use of document hierarchies in discussion forums and the like.

Additionally, there are security concerns: how are you going to map the security settings you take for granted in Domino, in SharePoint? Perhaps you're pushing selected document types to a specific document library, ready-mapped to fields, but how will you implement your readers and authors fields in an intuitive, useful way?

Do you have ready replacements for your *ad hoc* workflow processes? Easy integration with your email system? What about agents and script libraries: what will you do with that code and logic? Are your developers ready to take on the challenges of re-implementing much of this functionality in a system like SharePoint?



Considerations when migrating to LDC Via

LDC Via takes a different approach: you are pushing data from one document store to another which means removing many of steps listed above.

Notes design reviews? Extensive table design? Foreign and primary key work? Field mapping exercises? Losing hierarchies and doc-links? **No.**

LDC Via offers a new entry-point to your data

Clearly we can simplify the planning and design review stages, but what about the actual migration and deployment steps? And what about the end result: usable, useful applications?

LDC Via provides simple "data viewer" functionality as standard, like other data archival and migration tool-sets, but we take this much further. **LDC Via Lens** provides organisations with the means to configure screens for presenting data — indeed, this functionality goes further, providing the ability to build simple "forms" without a single line of code. Select which fields are presented on-screen, assign field labels, optional data types (including readers and authors), create drop-down lists, mandatory fields, computed items and more.

Even better, when you've finished configuring your forms, LDC Via Lens also provides configurable collection "filters", analogous to views in classic IBM Notes and Domino applications, with which to present your data.

Add mobile-ready user interfaces, out-of-the-box exports to Excel and PDF, and you can see how LDC Via Lens is a compelling option.

Your content, your way

Of course, migration isn't just about the documents. What about their content? What about rich text? It's important to try and preserve rich text with full fidelity. LDC Via will consider plain text, MIME content, embedded images, quoted printable content, and renders it all.

LDC Via also offers support for Notes-specific constructs like doc-links and document response hierarchies.

Security

When migrating to SharePoint, the Quest offering provides a number of options around provisioning Notes users in their new SharePoint environment, assigning security accordingly which then feeds through to a number of look-up tables in the SQL Server back-end.

Similarly, LDC Via knows all about Domino security. User accounts can be created and mapped (from LDC Via credentials to original Notes addresses) in one simple step. Document-level security is honoured with readers and authors attributes in the LDC Via platform, and users can be assigned (or denied) access to individual databases *en masse*.



LDC Via features

Applications: in addition to LDC Via Lens, LDC Via offers a number of standard “application templates”, i.e. simple, responsive web front-ends which use the LDC Via API to present content to end-users. These templates include a discussion forum, a document library, a read-only mail archive and a Teamroom-style application.

Comprehensive user management tools allow super-users to set up and edit users, assign access rights, and more.

A solid infrastructure for freshly-migrated apps built on standard, proven tools and platforms like node.js, MongoDB and Amazon Web Services — your technical staff will feel right at home.

Document store: no need for tedious mapping exercises or prolonged planning: simply use your data in a way that is both familiar and powerful.

Programmability: we offer an industry-standard RESTful application programming interface (API) which is well-documented and comprehensive: your developers can implement applications in the language of their choice using tools that are familiar and productive.

Simple user interfaces with LDC Via Lens: continue to extract business benefit from our legacy data and applications with the ability to create simple forms and views in LDC Via, no programming required.

A family of products: look out for KEEP.WORKS (in beta at the time of writing), a new NSF-based migration utility, with more to come!

SharePoint vs. LDC Via

SharePoint	LDC Via
Instance limits: SharePoint Online (Enterprise version) currently has a limit of 100GB per site collection and 300 site collections per account.	LDC Via (Enterprise) allows for an unlimited number of databases and documents, with a “soft” storage limit of 100GB which can be flexed to whatever is required.
Planning: in addition to mapping Domino fields to SharePoint / SQL Server, consider whether you should use lists (flat collection of defined records), pages (data documents) or libraries (collections of binary files) for each component being migrated.	No mapping required; documents map to documents. Documents using the same form definition are grouped together in “Collections”. Schema are not enforced, so documents in the same collection can have different fields, just as in Notes
Data types: if you elect to migrate to custom lists, be aware that this entails an understanding of over a dozen data column types: user, boolean, choice, lookup, etc. You will then need to map your required Domino attributes to these — potentially for each document and view type in your application.	All documents in LDC Via are stored in MongoDB’s binary JSON format (BSON). There are no enforced schema types, so fields require no mapping. Moving from Domino to Via doesn’t mean the massive context switch of moving from documents to tables.



Summary

LDC Via offers out-of-the-box applications, the powerful, configurable LDC Via Lens product and a number of cloud and on-premises deployment options.

Via also offers an industry-standard data store in MongoDB, a comprehensive application programming interface, user management tools, document-level security, and much more.

This adds up to a best-of-breed data migration tool for IBM Domino, and an application platform ideally suited for re-homing your IBM Notes and Domino applications and data.

Contact us

Web ldcvia.com
Email info@ldcvia.com
Twitter [@ldcvia](https://twitter.com/ldcvia)
Facebook facebook.com/ldcvia
Phone +44 (0) 20 3633 3009

About this document

This document was written by Ben Poole in February 2016 and substantially updated the following year.

This is version 1.4, published 4th March 2017.

About LDC Via

The LDC Via product line was founded in reaction to the clear need for a high-quality and simple mechanism to move data out of legacy IBM Notes and Domino systems into a modern, flexible platform.

LDC Via as an organisation grew out of the "London Developer Co-op", which itself was founded in 2008 as a small group of technologist with deep and lengthy experience in IBM Lotus Notes and Domino development, as well as other technologies including Java, node.js, .NET, PHP, Zend, react, MongoDB, MySQL, AngularJS, and more.

LDC Via Ltd is a company registered in England & Wales, no. 09209489.



Notices & copyright

IBM®, IBM Notes®, and IBM Domino® are registered trademarks of IBM Corporation in the U.S.A. and other countries.

Microsoft®, Microsoft SharePoint® and Microsoft SQL Server® are registered trademarks of Microsoft Corporation in the U.S.A. and other countries.

LDC Via™ and the LDC Via logo, as identified in this document, are registered trademarks of LDC Via Ltd. in the United Kingdom and / or other countries.

All other trademarks and registered trademarks are property of their respective owners.

© 2017 LDC Via Ltd. 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 LDC Via Ltd. ("LDC Via").