IBM Domino Application Migration and Decomissioning

1. Migration vs. Decommissioning

IBM Domino has been around for decades. Mid-sized to very large companies have built hundreds and thousands of applications around IBM Domino that support their business processes; many of those applications are business critical, e.g. because they store information that is subject to legal requirements.

But the IT world keeps changing. New technologies arise nearly every day. Companies need to continuously review their IT strategy in order not to fall behind.

Regarding IBM's Domino offering there are quite a few reasons why companies reassess their investments into that technology. Many of those companies already have decided or are in the process of deciding to move away from IBM Domino. They are looking at Enterprise Content Management (ECM) systems like Alfresco, OpenText Documentum, OpenText, SharePoint etc. to replace the applications they have built on IBM Domino technology over the last 15 to 20 years. Another option is to archive the information from these applications in an Enterprise Archive like OpenText's InfoArchive platform. And – what a surprise, this is not an easy job. IBM Domino as well as the applications built on top of it has unique features that the target platform might not support and that might be difficult to implement – even with huge customizing efforts.

From a pure technical perspective it is always possible to create a new application that is a one-to-one equivalent of the original IBM Domino solution and its logic based on one of these new technologies. However, this does often not make sense from a commercial point of view due to a large number of changes that would need to be made to the target platform – which in addition would mess up with any strategy to use standard software where possible.

So generally speaking there are two aspects that need to be distinguished when talking about application migration:

- Migrating the application and its logic
- Moving the data that the application maintains to the new platform

As outlined above, an application migration can be very time-consuming - and it's more likely to be than not to be. Therefore, the migration cannot be expected to be completed short-term. In addition, it typically involves huge financial investments. If the information that the application maintains needs to be retained, maybe for a long period of time, e.g. due to legal requirements, how should a company deal with this situation?

The answer to this question is »decommissioning«. When an application is decommissioned, the application and its logic are discarded whereas the data is maintained and kept in a way so that it is still accessible – at least for reading – even if the application that was used to create it is not available anymore.

> For the purpose of this document, migration is only targeting real changes of the underlying platform and not migrations from one version of an application to a newer one (update/upgrade).

2. Migration and Decommissioning Challenges

Projects to migrate or decommission applications are mostly driven by cost-saving considerations. At least that is what companies claim. In rare cases technical aspects might be the reason too, e.g. because an application does not fit into the company's overall IT strategy anymore or because an application has reached the end of its life cycle and will not be supported by the vendor in the future.

The major challenges to be addressed during an application migration or decommissioning project can be distinguished into two groups: Business and technical aspects.

The following chapters describe these challenges and come up with appropriate solutions to address them.

2.1 Business challenges

From a business perspective the major challenges of a migration or decommissioning project are:

- Costs involved
- Legal requirements to fulfill
- The time needed to complete the project

Of course, the weighting of each aspect might differ depending on the type of application that is due to be migrated or decommissioned. More details regarding each of these challenges follow.

2.1.1 Cost

The main cost drivers in migration and decommissioning projects are:

- Analysis of data that needs to be migrated/ decommissioned
- Selection of an appropriate technology replacement
- Customization of the target system to implement required functionality
- Definition, implementation, improvement and control of the migration process
- Ensuring high quality of migrated data
- Fulfilling legal requirements

2.1.2 Legal requirements

As already mentioned above, ensuring legal compliance throughout the entire migration/decommissioning process can be one of the costly elements in such a project. The entire process must be defined, implemented and controlled in order to make sure, documents are migrated completely. Any errors occurring need to be logged reliably and then be resolved.

In addition, the process needs to be documented end-to-end, outlining how requirements are addressed.

2.1.3 Time

Especially for applications containing a large number of documents, migration or decommissioning can take a long time.

Example: Calculation of processing time

Assuming an application has 250.000 documents and it takes 4 seconds to process one document from IBM Domino to the target system (end-to-end), this process would run for 11 days (250.000 documents * 4 s/document = 1.000.000 s = 278 h = 11 d).

In cases where applications are still being used actively, it's likely that new documents are created and existing documents are modified while the migration or decommissioning has been started already.

This imposes special handling of such applications and makes the entire process more complex.

2.2 Technical challenges

As expected, the technical challenges are far more complex than the business challenges. And as they largely depend on the type of application that is being migrated or decommissioned, the remainder of this document focusses on those that are relevant in the IBM Domino space.

From a technical perspective, the following challenges need to be addressed for IBM Domino applications:

- The number of applications can be very large
- Applications must not be changed to perform migration/ decommissioning
- Applications may contain (very) large numbers of documents
- Documents can be encrypted
- Document types in an application vary broadly
- Richtext fields cannot be easily exported and transformed
- Documents may contain attachments
- Documents can link to other documents
- Documents might have been versioned
- Documents might need to be editable in the new system

The following chapters will provide more information on each of these technical facets.

2.2.1 Number of applications

Over time, most companies using IBM Domino have created a huge number of different applications based on this technology. Even though not all of these applications contain critical information that needs to be maintained if the IBM Domino infrastructure is dismissed, the number of applications that are subject to migration/decommissioning can easily be hundreds or even thousands of different applications, if a company really relied on IBM Domino for a longer period of time.

The more of these applications exist, the more difficult a migration will be. Therefore, the technical approach to migration must be independent of the application itself.

In the context of IBM Domino application migration and decommissioning, an application is considered to be different from another application if it's based on a different application template.

2.2.2 Immutability of applications

IBM Domino applications are typically based on application templates. Templates can either have been provided by third party vendors ore created in-house.

The IBM Domino e-mail and document library template are examples of third party application templates.

If the technical approach for migration or decommissioning requires a change to the application, the change can either be made to the application or the application template, depending on whether it's required for all applications based on the template or just a few application instances.

For good reasons, any change to an IBM Domino application that is still actively used is considered to be critical by most companies' IT departments:

- If a change is made to a single application and the template that the application is based upon is modified subsequently, every individual change to the application needs to be re-implemented and tested after applying the new template version
- Therefore, changes are typically made to the application template rather than to the application itself in order to keep the application ecosystem maintainable. If an application template changes, the changed template needs to be applied to all the applications that have been built based on this template
- If a third party template has been used to create the application, any change to the template that a company makes in order to adopt it to its requirements might conflict with future changes that the third party vendor makes to the template. I.e. if the vendor changes the template, all adoptions made must be re-implemented and tested with the new vendor's template version. And this template then needs to be applied to all applications

2.2.3 Number of documents

As IBM Domino has been around for more than two decades, the number of documents stored in IBM Domino applications ranges from just a few to hundreds to thousands and even millions.

Efficiently extracting the documents and their related metadata can be challenging by itself because of the time needed to extract the information and transforming it so that it can be moved to the new target platform.

If the application is still being used actively, new documents can be created or existing documents can be modified while migration or decommissioning of the application has already been started.

Therefore the technical approach must be capable to run delta-migrations, i.e. detect new and changed documents and limit processing to those documents rather than requiring to process all documents again just because a single document has been created or changed recently.

2.2.4 Encrypted documents

Although document encryption is technically possible in IBM Domino applications, it is rarely used. Nevertheless, if an application contains encrypted documents, the technical approach used to export documents from that application must support on-the-fly decryption of documents during extraction.

2.2.5 Varying types of documents

In IBM Domino terminology, a document type is equivalent to a form that is being used to either gather or display the document's data.

A single IBM Domino application may contain a multitude of different document types, each with its own set of metadata.

When thinking of exporting documents from an IBM Domino database and storing that information in a new system, it's important that each document is exported in a way so that none of the information that makes up the document is lost on its way from IBM Domino to the new target system.

Although the representation of an entire document (metadata, attachments, links and versions etc.) in the new target system pretty much depends on the target system's capabilities, the approach for extracting documents from the IBM Domino application must be generic in order to be able to handle different document types without requiring special customization for each document type present in an application.

2.2.6 Richtext fields

Data stored in IBM Domino documents can have different data types.

While types like »date«, »time« or »text« etc. can be easily handled by any target platform, »richtext« fields are difficult to transform into a format that can be used reasonably in the new system.

IBM Domino richtext adheres to a proprietary specification. Richtext fields cannot be easily extracted and transformed into a format that can be reused in another system.

In addition, it might be especially important to actually preserve the richtext's appearance (text markup, tables, embedded images etc.) as it is shown in IBM Notes and keep the information editable in the target system.

2.2.7 Document attachments

IBM Domino documents can contain other files, either as »normal« attachments or as Object Linking and Embedding (OLE) objects.

Normal attachments can be easily extracted and stored as separate objects. If such an attachment contains a ZIP file, it might make sense to expand the ZIP file automatically and store the individual files contained in the ZIP rather than the ZIP file itself in the target system.

In contrast, OLE objects require special processing. There are no standard APIs in IBM Domino that allow for programmatic access to these objects and their extraction in a format that is usable outside of IBM Domino.

2.2.8 Linked documents

An IBM Domino document's richtext fields may contain links to other IBM Domino documents. The link target can reside in the same application as well as in another IBM Domino application.

These links between documents should be maintained in the target system wherever possible.

2.2.9 Versioned documents

Versioning is a feature that is used widely in IBM Domino applications, especially in applications like document libraries.

When extracting document from an IBM Domino application, it's important to preserve the version history and keep all information so that the evolution of the document from its initial creation to the last modification can be traced.

2.2.10 Editing documents after migration

If the IBM Domino application that a document originates from is still used actively and is supposed to rather be migrated than decommissioned, it might be important to transform the document in a way so that the document is still editable in its entirety in the target system.

3. Addressing the challenges

The business challenges as well as the technical challenges can make a single migration or decommissioning project a complex endeavour. But how can tens, hundreds or thousands of such projects be handled efficiently?

The following chapters describe how the »IBM Domino scanner for fme migration-center« helps overcome the various challenges described previously.

3.1 Universal usability and non-invasiveness

The Domino »C Application Programming Interface« (C-API) is the core API of IBM Domino. All IBM Domino and Notes features have been built on top of it.

The product uses the IBM Domino C-API to interface with IBM Domino applications. Therefore, any functionality the C-API provides can be leveraged.

The C-API is capable of:

- Authenticating and authorizing users
- Opening any database –either locally (filesystem) or remote (IBM Domino server)
- Reading and writing any document – whether it's encrypted or not – maintained by the application, given the user has the required permission
- Reading and writing any of the document's metadata (fields) independent of their data type
- Adding, extracting, replacing or deleting attachments (normal attachments as well as OLE objects)

All of these functionalities have been implemented in a generic fashion that is independent of the application, document types and metadata fields.

There is no need to modify any application to implement features required during migration or decommissioning.

3.2 Available export formats

The product supports a wide range of file formats to export:

- A document
- Its metadata (field contents)
- The attachments a document contains

The following sections will describe each of these formats and its benefits.

3.2.1 Documents

Depending on the business requirements (e.g. legal compliance) and the technical capabilities of the new target system, it can be necessary to export documents from an IBM Domino application in different formats.

The product supports:

- Domino XML (DXL)
- Internet Message Format (EML)
- Hypertext Markup Language (HTML)
- Portable Document Format (PDF, PDF/A-1a, PDF/A-1b)
- Rich text Format (RTF)

The product can generate any of them in any combination.

Each of these formats is generated using native IBM Domino functionality and will be described in more detail in the following chapters.

3.2.1.1 Domino XML (DXL)

Domino XML (DXL) is an XML based format. The XML schema has been defined by IBM and has been evolving continuously since its introduction with IBM Domino version 5.x.

The IBM Domino C-API offers the functionality required to export a document to a DXL file and store that file on the file system.

The DXL format preserves the entire document (incl. metadata, rich text fields, attachments, document links etc.). It is possible to restore a DXL file to an IBM Domino application. Although DXL is an editable format, DXL is well suited for storing a document in its entirety and ensuring none of its content is lost during migration or decommissioning, especially if the integrity of the DXL file is ensured otherwise (e.g. access control, read-only attributes etc.).

In addition, DXL can be used to create any of the PDF formats mentioned in this document (see 3.2.1.4). To do so, the DXL will be re-imported into the original application, opened with IBM Notes and printed to an appropriate PDF printer driver – all done in an automated process.

The combination of the DXL format together with a PDF rendition is a perfect match to preserve a document for long-term storage.

3.2.1.2 Internet Message Format (RFC 822/MIME/EML)

The Internet Message Format (RFC 822/MIME/EML) is mainly suitable for e-mail-like documents. It's based on Request for Comments (RFC) 822.

The ability to generate EML files from IBM Domino documents is available through the IBM Domino C-API. Documents can be exported as EML files and stored on the file system.

Like the DXL format, the EML format preserves the entire document. EML files store the document's contents in ASCII text and/or HTML together with any metadata and attachments that he original IBM Domino document had. However, it is not guaranteed that all text markup, OLE object, table and image information will be transformed consistently, which is quite obvious if the document's main content (e.g. the e-mail body) is transformed from rich text to ASCII.

EML files can be re-imported into a number of E-Mail applications including Microsoft Outlook, Microsoft Outlook Express, Mozilla Thunderbird and, since one of the recent versions, IBM Notes. This could be the format of choice to migrate IBM Domino E-Mail databases because E-Mails are still usable after the migration in e.g. Microsoft Outlook.

3.2.1.3 Hypertext Markup Language (HTML)

As many IBM Domino applications can be accessed from a web browser, the IBM Domino server is capable of rendering a document to Hypertext Markup Language (HTML) format.

This functionality is exposed through the C-API.

Another method for generating HTML is to generate EML (see 3.2.1.2) and then extract the HTML from the EML.

Generally speaking, it's not very likely that the HTML format is the first choice when thinking about migration or decommissioning of IBM Domino applications. HTML has several limitations (e.g. embedding or linking to attachments etc.) that cause it to be not well-suited for long-term storage. But HTML might be the desired format, if documents need to be displayed in a web browser.

3.2.1.4 Portable Document Format (PDF, PDF/A-1a, PDF/A-1b)

The Portable Document Format (PDF) defined by Adobe as well as its ISO-standardized format variants are best suited to preserve the full document context (contents, metadata, attachments) for long-term storage.

PDF renditions of an IBM Domino document can be either created from the document as it's stored in the IBM Domino application or based on a DXL (see 3.2.1.1) file of the document.

If required, the attachments that the original IBM Domino document contained can be embedded into the PDF file (see 3.2.3).

3.2.1.5 Rich Text Format (RTF)

Whenever it is required to keep a document editable once it has been migrated or decommissioned to the target system, the Rich Text Format (RTF) is the format to choose.

Although the C-API offers functionality to generate RTF, the product – by default – does not use this feature but rather transforms a PDF rendition into RTF. It has turned out that this approach yields maximized RTF quality and (nearly) eliminates the likelihood of losing information of the original document.

The RTF file can be edited in most word processing applications (e.g. Microsoft Word, OpenOffice Write etc.).

3.2.2 Rich text fields

Rich text fields are used in nearly any IBM Domino application somewhere to gather information from users.

Rich text fields can contain text with markup, tables, images, links to other documents, OLE objects and other pieces of information. Therefore, the contents of a single rich text field can be very complex.

Nevertheless, it may be important to preserve this content and all the information contained (incl. text's markup etc.) to comply with legal regulations when migrating or decommissioning an application.

Besides the ability to export a document in DXL (see 3.2.1.1) or any of the PDF formats mentioned above (see 3.2.1.4), the product offers the ability to export a single rich text field's content as a separate rich text file (RTF). If required, the fields to be exported to RTF files can be limited based on the field name by modifying the product configuration.

Exporting a rich text field's content to an RTF file allows for its content to be re-used in the new target system.

An RTF file can be edited in most word processing applications (e.g. Microsoft Word, OpenOffice Write etc.) or used in any control that's capable of editing RTF.

3.2.3 Attachments

Attachments that the IBM domino document contains can be extracted and stored on the file system. The product will store attachments in their native format, i.e. Microsoft Word (DOC, DOCX etc.), Excel (XLS, XLSX etc.) or PowerPoint (PPT, PPTX, etc.) or Adobe Portable Document Format (PDF) etc.

If required, ZIP files can be automatically unzipped, i.e. any file contained within the ZIP will be stored as a separate object.

In addition the product is capable of extracting OLE embedded objects from a document's rich text fields and store them as separate files on the file system.

The relationship between document and attachment is preserved so that it can be re-established in the target system, e.g. by putting all related files into one folder.

Last but not least, attachments can be embedded into a PDF rendition that was generated for an IBM Domino document so that the PDF file represents the complete IBM Domino document.

4. Conclusion

By using IBM Domino's C-API to control export of documents, metadata and attachments, migration-center uses a generic approach that does not require any changes to the application. In fact, it's independent of the IBM Domino application that is subject to migration or decommissioning, the type of documents maintained by the application and even the metadata (field) structure of the documents.

Supporting multiple export formats for documents and the ability to export normal attachments as well as OLE objects contained in rich text fields allows customers to configure the product in a way so that it helps address any of the challenges that might be involved with a migration or decommissioning project, including legal compliance.

fme group | Germany · Romania · USA
T +49 531 238540 · info@fme.de
T +1 203 6174250 · info@fme-us.com

created by
THiNK – e-solutions
Wienkoopstr. 3 | 69469 Weinheim

#301213