



Migrating an Enterprise to Adobe FrameMaker 7.0

Migration is typically motivated when a workgroup or enterprise needs to solve some set of business problems. The severity of these problems and the urgency to address them motivates the migration schedule and determines whether the migration is for a single project, a set of projects, a workgroup, or the entire enterprise.

Many organizations continue using outmoded tools and architectures long after those tools have been proven inadequate to new business challenges. In some cases, this happens because of budgetary concerns or because the organization's focus is elsewhere. Others, however, have the budget, the management interest, and the resources to move to a new platform, but delay for one reason—the migration task is considered to be too onerous to take on.

Business factors influencing migration

Adobe® FrameMaker® 7.0 software solves a wide range of business challenges at the workgroup or enterprise level. These are some of the business factors that drive migration to FrameMaker 7.0:

- **Current tools are inadequate.** They are not robust enough, slow users down, frequently crash, or corrupt documents. They cannot process complex information or large quantities of information well enough to meet the organization's size and content management, re-use, and repurposing requirements. They are not extensible to support new features, formats, or functionality.
- **Information has been re-engineered.** The information workflow and infrastructure (database, content management, and so on) have evolved such that the current tools no longer fit the management system or strategy.
- **Competitive advantage is being lost.** Current tools do not meet delivery requirements, including timely, on-demand information delivery, or customized or personalized content.
- **Efficiencies need improvement.** Reduced cycle times, improved time-to-market for information products, and marketing, product development, and manufacturing schedules are not being met. The organization cannot re-use information (create information once, distribute and package it many times and in many ways) and reduce the costs of information creation, maintenance, and delivery.
- **Quality is at risk.** Because tools are old, rare, and less user-friendly, and knowledgeable staff resources are more difficult to find, more errors are being introduced.
- **Regulatory compliance is required.** Current tools cannot tightly manage and control information to meet compliance requirements.

Moving from legacy information authoring and editing systems to FrameMaker 7.0 requires certain tasks and effort, based on best-practice migration methodologies. Let's take a look at these issues and develop a method to approaching migration.

Objections to migration

Migrating to a new platform does not need to be overly difficult or complex. With careful planning and execution, benefits arise because of the move to a better way of doing business. However, migrating to a new information authoring, editing, and management system

requires a commitment of time, effort, and resources. Approaching migration with the most detailed plan possible is crucial.

Adding FrameMaker 7.0 to a current solution

FrameMaker 7.0 software offers a wide range of compelling benefits to workgroup and enterprise information management: from template-based authoring to book-building features; from the ability to import many different file formats to multichannel publishing; from XML authoring and publishing to application integration capabilities.

Any migration planning must consider the current systems and workflow. Even in conjunction with an existing, successful infrastructure and workflow, the FrameMaker platform can almost always improve the content creation, management, and delivery process, as shown in the table:

How FrameMaker augments or improves on other popular solutions

Current solution	Example products	Current issues	FrameMaker 7.0
Authoring and publishing from the desktop	Microsoft® Office QuarkXPress Corel WordPerfect Corel Ventura	Fine tuned for basic word processing or layout and design	Combines content authoring, structure, management, layout and design, and publishing workflow capabilities. Provides template-driven authoring, ability to integrate content from many sources, and multichannel publishing capabilities .
Server-driven publishing	Interleaf XyWrite	Complex Require extensive configuration and development Use proprietary development and programming languages	Template-driven authoring and multichannel publishing. FrameMaker Server can be extended into any environment with its robust, C/C++ based API and development toolkit.
Database publishing	Oracle to a print mark-up system	Often complex and Difficult to manage and update	Integration options via MIF, XML, the Frame® Developer's Kit (FDK), template-driven rendering of content, multichannel publishing, and a highly configurable user interface.
Content-management system	Documentum	Often difficult for end users to utilize and configure	Template-driven authoring, integration capabilities, and powerful XML output. Integrates with popular document and content management systems, providing turn-key content management including version control and re-use.



Migration strategies

Every organization's toolset, workflow, and requirements are different, so every organization tends to have a unique migration strategy. Minimize disruptions and maximize chances for success by choosing an appropriate migration strategy. The three strategies described here are the most common. No matter which one is selected, migrating requires a commitment of time, effort, and resources that minimizes disruptions and maximizes chances for success.

Clean-slate strategy

Here the organization fully implements the FrameMaker solution for a specified prototype project. After the prototype project is fully integrated and tested, other projects are migrated into FrameMaker at the information-authoring phase. Interoperability or continuity with legacy processes and tools are not necessary with this strategy. Use this strategy when these factors are in play:

- Current tools do not meet needs for creating, managing, and delivering content.
- The current publishing solution is exclusively paper-based.
- A candidate project exists that is relatively segregated from the rest of the workgroup or organization.

Parallel-development strategy

In this strategy the new FrameMaker system runs in parallel with the legacy system, which is maintained as a backup until the new architecture is thoroughly tested and proven. Legacy content and workflows are then migrated to the new system. Here are some situations that call for this strategy:

- Current publishing processes are mission-critical; any loss of capabilities during the transition would be unacceptable.
- Current tools adequately meet existing requirements, but will not meet future requirements.
- No projects exist that are relatively segregated from the rest of the workgroup or organization.

Component strategy

Here the organization migrates only individual component areas of the information workflow, for example, content authoring or Portable Document Format (PDF) file production. Acute problems can be solved in these areas before a large-scale migration. Turn to this strategy in these situations:

- The workflow is highly structured.
- Specific individuals and departments have specific roles; for example, content contributors are responsible solely for content, and production staff are responsible solely for distribution.

Migration tasks

To execute any migration strategy, the organization must work through five sequential steps: evaluate needs, specify requirements, design and implement architecture, migrate legacy content, and train and deploy users.

Evaluate needs

Understand the workgroup or enterprise's need to migrate. Consider the forces driving the migration. Assess how the current tool's capabilities meet and fail to meet current and future requirements. Evaluate whether any requirements should be changed.



Specify requirements

These are set at the business level. Return on investment is the key motivator in this step. Specify the baseline functionality, performance, and stability requirements. In an enterprisewide deployment, consider the different requirements of individual users, workgroups, and the enterprise as a whole. Requirement considerations typically include toolset, workflow, delivery mode, content management, application integration, competitive strategies, production efficiencies, quality control, and regulatory and compliance issues. An example business driver might be: “deliver high-quality, personalized content to customers on demand.” The potential solution: A server-based publishing system, with FrameMaker 7.0 Server as the on-demand rendering engine for delivering PDF documents via HTTP over the Web.

Design and implement the architecture

Deploy FrameMaker, FrameMaker Server, databases, content management, and delivery. Include not only tool considerations, but also application integration, workflow, and testing. Also include design of authoring templates, and selection or creation of DTD and EDD for XML publishing. This step generally comprises four stages:

Architecture design—Design the architecture to address high-level business requirements. Determine whether the information architecture will be template-based (unstructured) or XML-based (structured). If template-based, the possibility of migrating to an XML-based solution in the future must still be open.

Component design and selection—Specification of the individual architecture components is driven by the organization’s current IT infrastructure (the enterprise database), and the strategies chosen for addressing the business requirements.

Component implementation—Design and implementation of the individual components of the architecture includes these stages:

- Supporting integration of components (for example, creating Frame® Developer’s Kit, or FDK clients to integrate FrameMaker with application servers for server-based publishing)
- Information modeling, including database schema and XML DTD design to support authoring and delivery requirements. With FrameMaker, the user or designer creates an XML application that associates templates, an XML DTD, read/write mapping rules, and optional processing plug-ins.
- Creating authoring and publishing templates. For an unstructured solution, this is valuable and necessary, so that a future migration to XML can leverage the templates designed now.
- Creating FrameMaker conversion tables to convert legacy unstructured content to structured content.

Component deployment—Scheduling component deployment, including time for testing and modification.

Migrate the legacy content

Legacy content can be one of the most important and resource-intensive factors in a migration process. The time required to convert a document depends on many factors, including the complexity of the document, its length, and the skill of the people performing the work.



Inventorying legacy content

A pre-migration content inventory examines the current information assets and helps the organization to make appropriate decisions about the migration strategy. How legacy content is treated depends on a number of factors, including these:

Location of legacy content—Where does the organization’s content reside (that is, its location in the organization’s file system, content management system, or database)?

Format of legacy content—Does the legacy content exist in its original application format? Is the original application still available and supported? Does FrameMaker recognize the original application format? If not, does the original application save in a more widely readable file format?

Some legacy content may be available only in a presentation format that cannot be easily edited and updated. These formats include hard copy (paper), TIFF images, and older versions of PDF that do not support saving as other formats. In this case, the organization should consider the services of third-party companies that specialize in migrating legacy content to current application formats.

Consistency of legacy content—If the legacy content exists in the original application format, was a common set of templates and/or styles used to create the content? Is the content consistently style-tagged? FrameMaker will generally recognize style tags in legacy application formats and convert those tags to FrameMaker paragraph and character styles.

Condition of graphics—If the legacy content includes graphics, are the original graphic files available? Is the original application that was used to create the graphics available?

Many word-processing programs destroy the integrity of imported graphic images by modifying the graphic file format for the convenience of the word-processing program. When migrating legacy graphics, return to the original graphics file format, upgrade that format as necessary to a current format, and import the updated graphics into the FrameMaker documents by reference. This solution retains the integrity of the original graphics file, allows further updates to each graphic (if necessary), and minimizes the size of the FrameMaker source files for better application performance and robustness.

Relevance of legacy content—Is the legacy content important for deployment of the new publishing solution? Will the new system focus primarily on delivering new content? If only a subset of the legacy content is required going forward, is it easy to identify that subset?

Target of legacy content—Are the legacy files “flat” or do they include structure information and metadata? Most unstructured documents have these characteristics in their original styles. For example, a “Heading” style is a strong indicator of a document heading.

A “conversion table” in FrameMaker helps convert unstructured content to structured documents, based on the documents’ existing paragraph and character styles. Depending on the XML publishing requirements, the results of the conversion table operation may be complete, or they may provide simplified XML documents that can be augmented with additional metadata in FrameMaker as part of the migration process.



Clean up the legacy content

Consider “clean-up” of legacy content in the original authoring tool to make migration to FrameMaker easier and faster. Expertise in that tool is probably currently available. Here are some issues to consider:

- How many different applications were used to create the legacy documents?
- What formats is the content in (paper, digital, other)?
- Do the documents have “implicit” structure that can be used to facilitate the conversion process? This includes common formats, styles, and the like.
- If the legacy documents do use some sort of style sheet, how consistently were the styles applied?
- How is FrameMaker similar to and different from the original authoring tools?
- How will FrameMaker change the workflow?
- What are the usage, training, and other staff issues?
- How will FrameMaker change the way content is managed?

Clean-up can be manual, automatic, or a combination of the two. These are some of the tasks involved:

- Scan hard copy pages to convert them to editable text
- Ensure that every paragraph and every specially formatted character has a style or tag.
- Programmatically tag content using a scripting or macro application (for example, Visual Basic for Applications [VBA] for Microsoft Word, LISP for Interleaf/Quicksilver, and AppleScript for many applications based on Apple Macintosh.
- Rename legacy tags or styles to FrameMaker paragraph and character style formats.
- Extract text, such as warnings and cautions, from graphics frames. Convert graphics from proprietary formats either via a graphics conversion application or by re-scanning from hardcopy.
- Identify symbols that do not survive the export because of a unique font encoding scheme in the legacy application. Replace them with unique character keys that will be changed back to the appropriate symbol in FrameMaker.
- Batch-export text and graphics.
- Reveal and hide text using effectivity (*conditional text* in FrameMaker).
- Determine how the legacy application supports autonumbers. If they are exported as hard-coded text, remove them before exporting.

Convert the content to FrameMaker

After clean-up comes conversion to FrameMaker:

- Programmatically convert the content using an off-the-shelf or custom filter (for example, the FrameMaker filter for Microsoft Word)
- Convert the content as a whole with in-house resources or with assistance from a consultant or a data-conversion company.

Clean up content after conversion

Virtually every conversion will require clean-up in FrameMaker. Repetitive tasks in FrameMaker can frequently be automated by a scripting language for FrameMaker called FrameScript, available from Finite Matters Ltd. (www.framescript.com). Post-conversion clean-up may be required in the following areas:

Untagged or rogue tagged text—Where formatting is missing or was misapplied, FrameMaker styles can be applied in batch mode using the global update options in FrameMaker, or individually.

Graphics—Imported graphics can be reimported into FrameMaker. However, graphics drawn in Interleaf/Quicksilver can be problematic. One technique is to export the legacy graphic as an



Encapsulated PostScript® (EPS) or PostScript (PS) file. Acrobat Distiller® can turn EPS and PS files into Acrobat PDF files, which then can be edited by Adobe Illustrator® software. Unlike bitmaps such as TIFF files, PDF files are vectors with smooth lines and text that can be edited easily. The Filtrix conversion filters from Blueberry also can be useful.

Tables—Ad hoc use of ruling, shading and cell straddles (merging) can fool converters, which prefer regular patterns. Table Cleaner, a FrameScript from Carmen Publishing, removes custom ruling and shading and performs other useful functions. For more information, see www.frameexpert.com.

Special symbols and equations—Reinstate the symbol characters that had been replaced by special character sequences during legacy clean-up to avoid exporting issues. Many filter applications do not support equations. FrameMaker includes an equation editor for re-creating equations in documents.

Train users and deploy the solution

If hands-on training is desired, choose a training provider and appropriate training classes. Deploy templates and DTD/EDD to authors. Receive and incorporate feedback.

Although an investment in training usually offers a rapid return on investment when increased productivity and improved business processes are considered, training can be a significant line item in a migration budget. By intelligently planning the organization's needs and timetable, the benefits of investing in training can be maximized.

A scaled training deployment is effective for many organizations. Begin by selecting a group of key people who have proven their ability to quickly learn new tools and processes, and who can support others during the migration process. Train this group early in the migration process, with advanced-level courses that will allow them to fully support the migration process. As the new publishing system is deployed, train more users, with courses appropriately targeted. For example, content authors who are not supporting template and DTD design functions do not need to be trained on those functions.

For more information

For more information on FrameMaker 7.0 and FrameMaker Server software, visit the FrameMaker 7.0 Web site at <http://www.adobe.com/products/framemaker/main.html>.

Adobe, the Adobe logo, Frame, FrameMaker, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Apple and Macintosh are trademarks of Apple Computer, Inc., registered in the United States and other countries. Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries. All other trademarks are property of their respective owners.

© 2002 Adobe Systems Incorporated. All rights reserved.

