

A Document Navigation System for the Paperwork Jungle

An electronic filing system won't make documentation any easier unless a company takes steps to organize and navigate that system.

Gerald E. Loeb, Jasspreet Singh, John Gregory Allen, Giby Raphael, Paul Buckley, Cesar Blanco, Francis J. R. Richmond, and Geoffrey D. V. Salter

Managing and locating technical documents is a serious challenge for medical device OEMs, which must comply with FDA's quality guidelines throughout the life cycles of their products. Documentation is at the core of everything a medical device company does. Quality systems can be summed up as "Write what you do, and do what you write." Imagine even one document per person per week. A company with 20 employees could generate more than 1000 documents per year.



A few decades ago, every organization had a well-staffed document control department and a lot of filing cabinets. Each had labor-intensive procedures to track released documents and recall obsolete documents. Now most documents are generated and circulated as electronic files, usually in a wide variety of formats (e.g., text reports, technical drawings, flowcharts, scheduling tools, spreadsheets, and pictures).

Many large organizations switched to complex and expensive online document management systems. However, most small organizations passed the responsibility to the individual employees generating and accessing the documents. Such systems depend on the logical thinking and continued presence of a few key employees; the systems quickly become overloaded as the number of employees and documents increase.

Searching for documents is a waste of time that tends to grow as new employees are added, experienced staff are lost, and the variety and number of documents increases. Documents tend to shift around in released, pending, and work-in-progress files. The files may have a logical directory structure, but that logic usually exists in the minds of the originators rather than the users of those documents. Seekers of documents in R&D and production departments are often frustrated and end up dragging their colleagues into the scavenger hunt while management fumes at the lost productivity. When the document is found, it's time for celebration. If it isn't found, it may have to be reconstructed from notes—a tedious process—and there may be serious problems with audits and regulators.

Many such episodes and problems led one organization to develop a software product that maps electronic paperwork. It is particularly useful for start-up businesses and small to medium-sized organizations.

It is important to have a single source for released documents to ensure that all stakeholders have access to the most current information. The document navigation system (DNS) developed by the Alfred E. Mann Institute for Biomedical Engineering at the University of Southern California (Los Angeles) is based on a simple set of requirements. The primary requirement is that the system enable intuitive and easy retrieval of information. Other requirements include the ability to integrate different types of documents and files (e.g., drawings, pictures, documents, spreadsheets) with minimal redundancy and the ability to navigate easily within and between the documents.

Because most of the information in such systems is confidential, access should be controlled. Most users should have only read-only access, while a select few can update and release the documents. Finally, a DNS should not make any assumptions nor impose any limitations on

existing systems for computer operation, document generation and control, file storage and backup, local- and wide-area network access, etc.

It is important to note that the system described here is a document navigation system, not a document control system like those currently on the market. Document control systems are very useful for tracking changes introduced by multiple authors who already know which documents exist and where to find them. A DNS is designed for personnel who need to find and read those documents. In some cases, personnel don't even know which documents exist, much less what they are called and where to find them.

Examining a DNS

A DNS uses pictures in the form of icons to provide an intuitive skeletal structure to the architecture of the document organization. The manufacturing sequence—the assembly tree—of a device is probably the most natural and logical image that can provide the structure for documents. The tree is a standard part of a device master record, and everyone in the organization who needs access to the documents likely already knows it well. Therefore, the tree can provide a top-level map on which all documents are linked. As mentioned earlier, a DNS is not to be confused with a document control system. A DNS serves as a graphical aid to display the latest versions of released documents using a graphical user interface.

A company also needs a document control system to keep track of documents by type, by team, by approval status, etc. All of a company's documents should reside in directory structures that can be organized in any way that makes sense to the personnel who draft, approve, and distribute documents. In contrast, a DNS is for the end-users of documents: the scientists, engineers, assemblers, technicians, and support personnel who need quick, reliable access to current and correct information. These individuals may not understand the document directory structure, but they do understand the product.

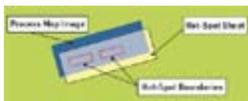


Figure 1. (click to enlarge) The graphic interface starts with a jpeg image depicting the process flow. A DNS allows an administrator to create a hot-spot sheet that causes a cursor passing over a spot to turn into a hyperlink to an outline of linked file names in a database.

A DNS is a Web-based application that uses any standard graphics package to create an image of the process map (i.e., assembly tree or process flow diagram in any graphical file format). This image is then overlaid with an invisible hot-spot sheet that is used to create links to various documents related to the process item (see Figure 1). It differs from more-casual use of links in that the creation and maintenance of those links is reserved for a DNS administrator. The released documents are actually located in a controlled-documents repository maintained by that administrator, and they are linked to the process map through a MySQL database. The database allows users to collect, prioritize, and organize multiple types of documents that may be associated with each process step and its hot spot (e.g., drawings, specifications, standard operating procedures, data-recording forms, and equipment maintenance logs). However, the source documents never move from their released file directories.

Functional Design

A DNS has two modes: user and administrator. Users have read-only privileges and can access the assembly tree and all of its related documents via the Internet or an intranet. Administrators have write privileges to the DNS database and can create, delete, edit, and update all of the links. The administrator acts as a document controller, ensuring that the latest released version of each document is available to the users. Each document is formally reviewed and signed off by the approval committee before it is released to the controlled-documents repository, which is a file-

server directory with restricted write-access. A DNS administrator then links the documents to the assembly tree.

Note that both the end-users and the DNS administrator need only read-access to the controlled-document repository. Organizations can have multiple products, each with its own assembly tree and DNS database, but they can freely share any or all files in the controlled-document repository. By writing the DNS system in HTML, it can be placed in a password-protected Web page, so users can access documents from any location.

Part of the assembly tree for an injectable neuromuscular stimulator is shown in Figure 2. This particular tree includes many mnemonic cartoons of components, as well as color-coded part and fixture numbers. The illustration of the tree could, however, be made as simple or complex as desired because it exists separately from the DNS. It can be created in any graphics package as long as it can be saved in jpeg format.

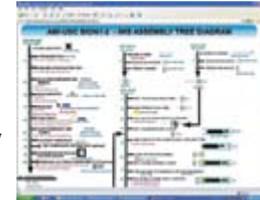


Figure 2. (click to enlarge) Web browser view of the top half of an assembly tree for a neuromuscular stimulator as an example of a Class III implanted electronic device produced according to FDA GMPs. Such an illustration can be created in any graphics package that can be stored in jpeg format.



Figure 3. (click to enlarge) The assembly tree shows the outline of its structure. Associated documents and the color-coded hot spots are overlaid on the tree itself.

An assembly tree naturally tends to be hierarchical, so it resembles an indented, color-coded outline in the DNS (see Figure 3). The top level is the entire assembly tree for the product (shown in red), followed by individual subassemblies (shown in orange), steps in the manufacture of each subassembly (shown in green), and components that are added at each step (shown in dark blue). Each level of the outline can be associated with any number and type of documents (shown in light blue), which then can be located using a typical directory-browsing window.

New items added to the outline are denoted in bold type until they are associated with a hot spot on the drawing. A hot spot is created by right-clicking on the outline item and creating a rectangle at the desired location on the drawing, using the typical click-and-drag drawing tool.

Several features facilitate maintenance of a DNS throughout the inevitable cycle of changes to the assembly tree and related source documents, such as the following:

- If the process image has been changed (e.g., to make room for a new step), related hot spots can be grouped to move them and their associated links together.
- When a document file name is added to a DNS outline, the DNS database links it to any other instances of the same file name. If a file name is changed anywhere (e.g., to point to a revised version of the document), the DNS software asks whether the administrator wants to change it in all instances in the database.
- The outline view can be expanded or collapsed by double clicking on the plus (+) or minus (–) symbol next to each entry according to the usual convention. This makes it possible to see the overall structure of the tree even after it has accumulated many levels and associated documents.

A user's view of the assembly tree is identical to the original graphic (e.g., jpeg file shown in [Figure 2](#)). However, the cursor changes to a link symbol as it passes over the invisible hot spots. If a user clicks on one of these hot spots, a document access window opens, as shown in [Figure 4](#). It lists all documents associated with that level of the outline. Clicking on any one of these document names opens a window in which the document is viewed in its native application. That is, a standard operating procedure written in Microsoft Word opens a Web page in Word, a scanned document in pdf format opens in Adobe Acrobat, and a mechanical or electronic schematic drawing opens in the native CAD package in which it was created. This permits the document to be printed, excerpted, or edited and saved as a new document in another directory, but not overwritten in the write-protected directory holding released documents.

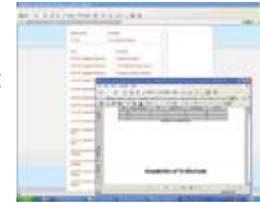


Figure 4. (click to enlarge) As users position the cursor over the assembly tree shown, the cursor changes to a link symbol whenever it passes over a hot spot. If a user clicks on a hot spot, the appropriate page of the document outline appears, and the user can click to select any document related to that level of the outline. The document is then retrieved from the controlled-documents repository.

The list of documents in a user's outline view is ordered according to the extension, indicating document type rather than the order of entry into the outline. The database contains a list of recognized extensions ordered according to the desired display priority; unrecognized extensions are displayed at the bottom. A DNS ensures that all documents listed in the outline are relevant to that particular part of the assembly tree, but it helps if users can always find various types of documents in the same order. The displayed list of locally linked documents may be long and diverse.

Document file names are often arcane, particularly if they are based on numbering schemes or acquired from component vendors. With a DNS, users don't need to know the document names at all.

Validation and Security

This particular DNS was designed to be below the radar of software validation. It is not a product that will ever be seen by patients or clinicians. It is not used to edit or save actual documents, nor does it provide storage or backup facilities. Instead, it simply provides pointers to documents generated, stored, maintained, accessed, and backed up by the document control, storage, and access system used by the organization. Nevertheless, DNS administrators need a standard operating procedure to ensure that all new or revised documents are linked promptly, correctly, and at the most useful level of the assembly tree outline.

The role of the DNS administrator is important, but the information and procedures needed to do this job should already be part of the company's existing document control system. The most important piece of information is an accurate and current assembly tree in the device master record. If the organization lacks that, then its problems run deeper than document navigation.

It may be valuable for product management teams to periodically review the accuracy and completeness of the company's DNS for a specific product. One convenient and useful time for this exercise is during meetings of the product quality review board when it is considering substantive changes to product design or manufacturing. A DNS provides a useful overview of all documents that need to be reviewed and a way to access those documents quickly during the meeting itself. It is particularly valuable to remind the board about all of the documents that would need to be revised to reflect a given change. Inconsistencies among documents are a frequent cause of process control problems. For example, changing the dimensions of a component may require changing the incoming inspection procedure, the dimensions of the fixture that holds it, the dimensions of components with which it interfaces, the cost accounting spreadsheet, etc. A DNS administrator can create a shadow database that includes both the new and unchanged pointers to all of these documents for the review process. The administrator can then quickly swap the old database for the new database after approval is received.

A DNS restricts unauthorized access to its resources and defines different privileges at both the user and the administrator levels. The right of entry to DNS Web pages from a remote computer is controllable using a Basic Authentication scheme that is built into the HTTP specification. A dialog box requests a user name and password and then lets approved users in. Each user must have an account on the Windows NT machine where the Microsoft Internet Information Services (IIS) Web server runs. Access permissions to the documents can also be assigned individually.

The basic drawback of this scheme is that user accounts and passwords are being passed across the Internet in plain text. IIS also has another authentication scheme called NT Challenge and Response, which uses some variety of encryption with the password. However, a DNS does not have its own security mechanism. Rather, it is designed to work with any security system that has been implemented by a network administrator.

The DNS administrator part of the software has the interfaces to edit the template, create hot spots, and link released documents. It should be installed only on the administrator's computer. The DNS administrator should work closely with the network administrator to control the access rights and prevent unauthorized access to his or her computer. This depends on enforcement of any computer security policies implemented by the company.

Implementation

The Mann Institute DNS was coded in Visual Basic. Microsoft's ASP.NET Web-application interface supports many languages, including VB, C#, and Perl. Microsoft's development environment software, Visual Studio .NET 2003, provides a graphical user interface to create these applications. All HTML code is generated automatically by the application development environment and stored in ASPX pages that open in any Web browser. Other tools used include MySQL (Version 5.0, Enterprise edition), MySQLDirect.NET data provider (2.80, Core Lab. MySQL.dll), and IIS Web server (version 5.1).

The DNS consists of Visual Basic modules called DNS Admin (installed with the database on a secure computer in the local-area network) and DNS View (downloaded and launched by the IIS Web server), which support the administrator and user modes of use, respectively.

Conclusion

The intuitive graphical user interface for a DNS makes it easy for new employees, consultants, and managers to get both the big-picture view of a product and the details that they need during their daily work. It is also a useful tool for supporting team conferences and design review meetings. For example, if a question arises about a particular process or component, it is easy to drill down to the current source document needed to inform the discussion. More importantly, the team can see and immediately access all the related documents.

One of the most frequent sources of confusion and errors is the unwitting introduction of factual incompatibilities between documents of different types. But a DNS provides a solution to this problem: it shows users all documents that are functionally related to each other.

The ease of use of a DNS simplifies the problem of document control in the organization. It reduces the temptation of personnel to copy and maintain personal hordes of frequently needed documents, which inevitably become obsolete and uncontrolled. If users go to the DNS whenever they need information from a document, they will automatically find only the current version of the document.

A DNS can be used for and organized around structures other than the manufacturing assembly tree. Companies often use graphical structures to illustrate such items as reporting structures for

clinical trials or project budgets. The DNS can create linked overlays so that users can find various supporting documents.

Linked documents have long been touted as the way to help readers move quickly to the additional information that they need. However, links point to one specific version and location of a document, which may need to be updated or moved to comply with document control requirements. Without a way to build, locate, and manage all of the links centrally, readers will unwittingly stumble through obsolete or broken links. A DNS provides a central database and management tools to create and maintain hyperlinks according to the natural structure of the product.

Acknowledgment

The DNS cited here was developed by the Alfred E. Mann Institute for Biomedical Engineering, University of Southern California (Los Angeles). The programs and documentation regarding their installation and structure can be downloaded as freeware at <http://ami.usc.edu> under Projects/BION/Publications/DNS Software. Please note that this is not commercial software, and the institute makes no warranty regarding its functionality or support. If this software is downloaded, please ensure that it complies with local IT policy.

If external users provide enhanced versions with appropriate documentation, they may be made available through this site, but there are no plans to support formal open-source development. Contact Paul Buckley for additional information at pbuckley@usc.edu.

Gerald E. Loeb, M.D. is professor of biomedical engineering, director of Medical Device Development Facilities in the Dept. of Biomedical Engineering, Viterbi School of Engineering, and Alfred E Mann Institute for Biomedical Engineering at the University of Southern California (USC; Los Angeles). He can be reached at gloeb@usc.edu.