

Savvy is an Internet based, document management system. Savvy allows users to view any document located on any server using just their web browser.

Savvy organizes documents into folders. All folders on Savvy are virtual folders so document can be in more than one folder even though it is in only one location on the computer.

Savvy organizes users and documents into Domains. Savvy can support unlimited domains, unlimited users per domain, unlimited user groups per domain, and unlimited documents per domain. Savvy users can be placed in groups to make security assignments easier.

Each Savvy domain supports both public and private access. Documents in the domain can be assigned for public viewing or protected and reserved for individual users or groups.

A Savvy domain can be spread across an unlimited number of physical web servers. These web servers, known as Remote Document Servers, can be anywhere on the Internet. Savvy organizes the documents from all of these Remote Servers into a clear and concise entry point.

Savvy has a very strong document searching facility. Users can search for documents across all remote servers at once!

Savvy is an ASP.Net web application, which consists of two major components - the Savvy Document Control Server (DCS) and the Savvy Remote Server Manager (RSM). Document requests from the user are validated and packaged by the Document Control Server and sent to the Remote Server Manager. The Remote Server Manager then delivers the request to the DCS. The DCS then forwards the document to the client's browser.

The Savvy Document Control Server

The Document Control Server is the focal point of Savvy. All Savvy Users talk directly to the DCS. The DCS maintains each user's login credentials and permissions and is responsible for user authentication and authorization.

The DCS also maintains the Document Profile Database. This database stores all documents, users, and Access Control Lists for a given domain.

Users access a Savvy domain using their browser. If the domain has public access enabled, DCS greets the public user with the Savvy home page. The DCS then presents any documents the public user is allowed to view in a categorized tree view in the left pane of the home page. If public access is not enabled, DCS presents the user with a login page.

DCS validates the user's credentials then prepares the tree view with the list of documents the user (or user's group) has permission to view. DCS will not allow the user to view a document or folder unless the user has permission to view it.

When a user requests a document, the DCS determines which Remote Server contains the actual document then fetches the document from that server. The DCS then delivers the document to the user.

DCS also controls document searching. Documents can be searched by content or profile. Content searches take place across all remote document servers. DCS collects the search results from each document server and consolidates the results into a single search tree.

The Savvy Remote Server Manager

The supporting component of Savvy is the Remote Server Manager or Guardian. The RSM is an XML Web Service, which is installed on each remote document server. The main responsibility of the RSM is to deliver a document to the DCS. The RSM is protected by NTFS security. The login information required to access the RSM web service as well as the IP of the computer running the RSM is placed into the Savvy Configuration database at the DCS.

RSM also interfaces with Microsoft Index Server to perform document searches based on the criteria it receives from the Document Control Server.

The documents are stored on the remote document server in Windows folder designated for that purpose. This folder is known as the Document Root. The Document Root can be the root of an entire drive.

When a Savvy user requests a document, the DCS logs in to the RSM using the userid and password stored in the DCS. Once connection is established, the DCS requests the document. The RSM then encrypts the document and sends it back to the DCS. The DCS then decrypts the document and forwards it to the requesting user.

This entire process is totally transparent to the user.

Savvy Security

Since Savvy is Internet based, special attention has been paid to keeping both documents and remote servers secure.

First, a Savvy user must log in to Savvy before he/she can view the allowed documents. A user that does not have permission to view a folder or document will not see it in any way. That document or folder will not exist for that user!

When a user requests a document, he/she is not simply redirected to the document server. To do so would allow the user (or someone with access to the user's computer) to bookmark the page and access it later. The user would also have to have a user id and password to that server. Should the user's rights to view that document be terminated, they would still be able to access it through their bookmark.

To prevent this, Savvy DCS acts as a "go between" or "middle man" between the user and the RDS. Only the DCS knows where the RDS is located and only the DCS can log into it.

The User *never* accesses the RDS directly!

Savvy DCS fetches the document and forwards it to the user. The user thinks the document came from the DCS. They never see the RDS's url at all.

Savvy DCS does this by using a technique known as url rewriting. The DCS has a special folder named rdf. Nothing exists in this folder. But all documents appear to come from this folder since the DCS will map a given request to it. Attempting to access the folder through a bookmark will fail.

Using SSL

Savvy encrypts all user credentials sent over the Internet. So even if a web server is set up for Basic Authentication the login credentials are secure.

The documents themselves are encrypted between the RDS and the DCS. Since Savvy is a web application, certificate type SSL is used to encrypt traffic between both the client's browser and the document store.

Savvy's security systems let your documents be viewed by only those users that have permission to view them.