1

Official Records

Records management is driven primarily by regulatory compliance and the need to reduce the risk of exposure to legal liability for improperly managing information. The prospect of paying significant fines for not adhering to an increasing array of regulatory requirements provides a strong incentive for companies to implement comprehensive records management solutions. But there can be equally significant costs associated with implementing those solutions that must also be considered. There is the potential for lost productivity as knowledge workers spend more and more time focusing on records management issues. There is also the potential for increased IT costs as additional time and energy must be devoted to building and maintaining the records management infrastructure.

That infrastructure has requirements as well. Not only must it support the identification and handling of *official records*, but it must also provide the same tools for collaboration, approval, and workflow that other knowledge workers employ in their day-to-day work. Compliance officers are knowledge workers too, and the job of managing official records can be even more daunting than creating them.

> *Microsoft has published a comprehensive records management planning guide that is essential reading for anyone involved in the development of records management policies and procedures, regardless of whether you are using MOSS as an implementation platform or not. You can download the guide from* `http://go.microsoft.com/fwlink/?LinkID=92720`.

What Are Official Records?

Don't you hate it when you ask someone a question and they respond with the generic answer, "that depends"? I often find myself thinking, "This person obviously doesn't know the answer to my question and is just stalling for time so they can come up with something intelligent to say!" You see where I'm going with this, right? When thinking about records management, the first question that often comes to mind is, "What is an official record, anyway?"

When does an ordinary document become something more? When does it "officially" become something you have to treat specially because of some legislation or corporate policy or other business process that may govern what you might otherwise do with it? Well, I hate to do this to you right off the bat, but the only answer I can give you is, "that depends." Let's dig a little deeper, and you'll see what I mean.

To answer this simple question, there is a lot to consider. Is there something about the document itself that determines whether it is considered an official record? In the simplest case, that determination might flow from a special set of values in one or more document properties. In other cases, it might come from the presence of a particular kind of content within the document. For example, the document might contain sensitive information that only certain people should be allowed to see. Ultimately, particularly in the context of highly collaborative environments like SharePoint, the final determination of what constitutes an official record may have as much to do with how a document is used as it does with the document content or with the metadata associated with it.

Traditional records management systems are focused primarily on archival mechanisms. We could characterize them as *location-based*, where the physical location of the document is the dominant consideration.

If physical storage were our only concern, then building records management solutions would be fairly simple. Assuming we had a consistent platform for associating metadata with a file, we would basically only need an established process for storing the file in a secure repository so we could find it again easily. Obviously, we would also need to apply some discipline in categorizing the metadata according to a given set of rules, but this would be a relatively straightforward process.

The more challenging scenarios involve concurrent processes that may have nothing at all to do with the policies or rules we are trying to enforce. This is where SharePoint distinguishes itself significantly from other systems. Dealing with official records in the context of dynamically changing metadata without interfering with critical business processes represents a quantum shift in the way we think about official records and the software components needed to manage them effectively.

Core Records Management Principles

What are the core principles of records management? Here, I'm talking about the essential characteristics of any records management system. At a minimum, there are four: confidentiality, information integrity, adherence to policy, and auditability. If any one of these capabilities is missing, then that system does not provide the necessary control points for ensuring that a given organization is meeting the regulatory requirements driving the decision to implement the system.

In the SharePoint environment, we can add one additional core requirement: high availability. This gets back to the notion that whatever controls you put in place for managing official records should have minimal impact on normal business operations. For collaborative documents that support multiple business processes, it means that you must implement solutions that do not hinder those business processes or increase their cost.

Now look at these principles of records management more closely:

❑ **Confidentiality** — Confidentiality requires that the records management system must ensure that strict access controls are maintained for any official record so that only those persons and groups with appropriate permissions are able to view the record. Any deviation from such access controls must also be properly monitored and recorded.

❏ **Information Integrity** — The records management system should provide a way to check the integrity of a record's metadata as well as its content. This could require the enforcement of rules that govern the range of property values that are considered *valid* for a given set of metadata fields. The system must also ensure that neither the content nor the metadata associated with records is altered after they have been placed into the repository.

❏ **High Availability** — Official records must be available at all times to support processes that may or may not be related to core business functions, such as litigation support or compliance research. This requirement of high availability often underscores the need to decouple the Records Repository from other enterprise information stores so that any request for official records is not tied to critical business processes.

❏ **Adherence to Policy** — *Adherence to policy* means that there are rules that govern how particular types of records must be handled by the system. For policies to be implemented effectively, they must be defined clearly and must also support administrative proof that a given policy was followed for a given record instance.

❏ **Auditability** — The auditability requirement means that there must be an efficient and secure mechanism for keeping track of everything that happens to a record. This typically includes changes to the way in which the auditing itself is implemented. Thus, audit records are also treated as official records. Consequently, the records management system must provide tools for ensuring that the auditing features are also tamper-proof and that the audit records are securely maintained.

Content Modeling

What are the essential characteristics of content that enable effective records management? Is there a methodology we can use for deciding what metadata to associate with a given content element? Ultimately, we want tools that not only enable us to build detailed models of our metadata design, but also to bind the resulting metadata to our code consistently and in a reusable fashion so we can avoid having to build the same model repeatedly.

What we have today is a pretty haphazard definition of what metadata consists of and how it should be defined. There are different kinds of metadata coming in from legacy document management systems and from unstructured documents. The properties attached to those documents haven't necessarily been defined in any consistent or rigorous manner. Consequently, when we're building records management solutions in SharePoint, we run into the problem of matching columns to properties and making sure that each document being managed by our solution has appropriate data in the right places. This can be a big problem, and it can get very complicated. There are third-party tools cropping up that purport to provide mechanisms for mapping properties, and thus the concern is not so much about the actual mechanics, but about the methodology used for figuring out which properties are important for a given scenario and deciding what we're going to map those properties to.

Another way that metadata is being determined now is that there are often many different copies of the same document floating around in the enterprise. In a perverted kind of way, the more duplication there is of a particular document, the more we could say that document may increase in value. Because it is being used so much, it must have more value than other documents. It becomes a sort of *high-value target* as a starting point for further analysis. By examining frequently copied documents, we may develop a rudimentary understanding for how the document is being used at different times. As an example, let's say that we have an annual report that is being created consistently every year. But then

at some point, new management takes control of the company and they need new information, and the annual report morphs into a different kind of document that includes additional reports. Another example might be an expense report that has new expense items that must be included. Whenever a document that is attached to one business process is applied to different business processes, the result is that the metadata gets messy very quickly. Certain properties support either the original process or the new one, while other properties support both.

Generally, there is no systematic method for determining what metadata is needed for a given business purpose. Yet the paradigm keeps evolving, and we're moving forward with this content explosion. We have no choice but to move more and more content into SharePoint based on a pretty informal methodology. To deal with this, it helps to think about how the metadata is going to be used. If it is simply for classification and categorization, it may not be as important to have formal methodology for figuring out what the metadata should be. But if it is going to be used to drive business process automation, it becomes vitally important to have a methodology so that we can consistently determine or derive the correct metadata because we need to coordinate multiple processes and the activities being performed by people connected to those processes. Where is that coordination going to come from? It's going to come from something attached to the document or to some other object that describes the document. In a traditional document management system, we might have a database record that includes this metadata. In SharePoint, we can attach the metadata directly to the document, associate it with a list or content type, or embed it into a workflow activity.

Understanding the Content Life Cycle

The content life cycle provides a useful model of the way humans interact with information, and it can be represented as a simple sequence consisting of four phases: creation, review-and-edit, publication, and final disposition. Depending on the type of content being modeled and the way it is used, the review-and-edit phase may be repeated as often as needed.

It is worth noting also that this is a recursive model, meaning that the entire sequence can be embedded within any individual phase to drill deeper into the analysis of the content-driven workflow. For example, the process of creating an annual report could be modeled as a sequence of steps starting with the gathering of initial metadata to create the document instance, review-and-edit of the content and underlying assumptions, approval and publication of a series of drafts, and then final disposition to a shredder or long-term storage repository.

But the process of gathering the metadata could itself be modeled in the same way. This is because what we call *content* depends on many factors, not the least of which is the context of the analysis. Figure 1-1 illustrates the basic sequence.
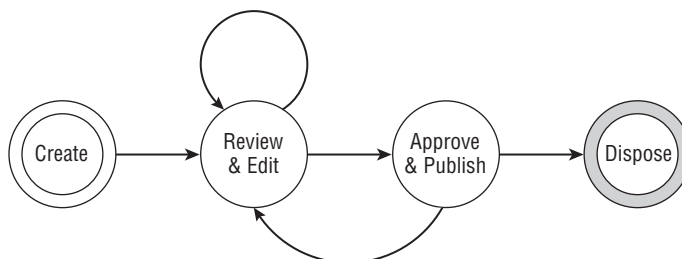

Figure 1-1: Typical content life cycle.

The content life cycle can help a lot when building ECM solutions because it provides a context within which to understand and tease out the requirements needed to deal with chaos. The content life cycle informs further analysis of those requirements, even though our current tools for creating content provide no consistent data model that can be leveraged to answer basic questions about the information we use every day to perform common business functions. The key lies in the fact that content drives all business processes. If we can clarify just what kinds of interrelationships tend to help or hinder those processes, we can define and identify critical metadata that can then be used to manage the transformation of content from one stage to another.

At each stage of this life-cycle model (creation, review-and-edit, publication, and final disposition), different tools may be used to manage the production or consumption of information at that stage. Having the right tools at the right stage can have a significant impact on whether a given business process is successful. It can certainly influence how quickly an organization can respond to changes in the underlying information. Therefore, it is useful to explore ways to leverage whatever can be known about the life cycle of a particular kind of content to understand how that content relates both to the individual producers and consumers of that content and the environment surrounding how that content is used.

Consider what happens during the content creation phase. Whether using a server or client application, the way that content is created is driven at least in part by the role being played by each person who interacts with the content or with the initial metadata used to generate that content. As an example, if a team is producing a periodic report such as an annual report for a company or a quarterly report for a department, then the appropriate tools must be available for each team member to create the required metadata before the report can be created (whether it is generated or being written by hand). Once it is created, it then moves on to the next phase in its life cycle and may require different tools at that point.

What's interesting is that the role being played by each team member as well as other essential characteristics of the content itself help to define what tools are needed at each stage. This intersection between the content life cycle, the roles of each content producer and consumer, and the business process to which the content is being applied defines a "nexus of opportunity" that can be exploited to achieve new levels of business efficiency. The trick will be to find an effective technique that pulls all of the required elements together in a way that engages all stakeholders. First, we'll lay out some high-level goals for our content modeling technique, and then we'll look at role/activity modeling, which addresses these goals.

Content Modeling Goals

Goals that you should keep in mind as you develop a content modeling methodology generally include the following:

1. Understand the essential characteristics of content that can enable effective management and control.
2. Develop a methodology for deciding what metadata to associate with a given content element.
3. Develop tools for binding content metadata to code in consistent and repeatable patterns.
4. Build an effective strategy for using content to support business process automation.
5. Be precise and not require heavy IT skills or sophisticated modeling tools.

Ideally, the modeling methodology should not require heavy technical skills. The target audience will include both developers and knowledge workers who typically understand business processes better than IT does. We don't want to introduce a modeling technique that is overly technical. At the same time, it has to be precise enough to support the kinds of quantitative analysis we need for software component building in order to respond to the business requirements.

The final goal is that we don't want to have to use sophisticated modeling tools. We should be able to create these models easily using something no more complicated than Microsoft Excel. We should also be able to create visual models easily that relate directly to the actual modeling activity using something like Microsoft Visio so that we can easily share the model with knowledge workers, developers, and architects.

The Role/Activity Modeling Technique

We use role/activity modeling to identify the content that drives specific business processes, and then we classify the content so that we can map it to different parts of the solution. The remaining content is grouped for highest efficiency, such as by predominant role, security requirements, source location, and so on. Finally, we identify and build domain-specific components (such as timer jobs, event receivers, and workflow activities) around those content elements. This results in a set of components that is more tightly coupled to the business process being automated and greatly simplifies the construction of solutions around that business process.

> *This technique is particularly useful for workflow development because the components you identify can be implemented as custom workflow activities that can then be added to SharePoint Designer to enable business analysts to easily add domain-specific workflow support to any SharePoint portal.*

The role/activity modeling exercise begins with a guided discussion to identify the major roles that will participate in the solution and the primary responsibilities that will be assigned to those roles. Then we walk through a very informal, but still structured analysis to determine what the main activities are that are required in order to fulfill those responsibilities. Within each of these activities, we then determine what information is essential to performing the tasks we have enumerated. As it turns out, this is a very useful analysis when dealing with any kind of portal because in the portal environment, you are presenting users with tools for performing tasks that fulfill responsibilities to which various roles have been assigned. And so that maps pretty nicely into, for example, an Active Directory permission-granting environment wherein you are providing certain tools to certain individuals and information is constantly flowing into and out of the system. Figure 1-2 shows the steps of the technique.

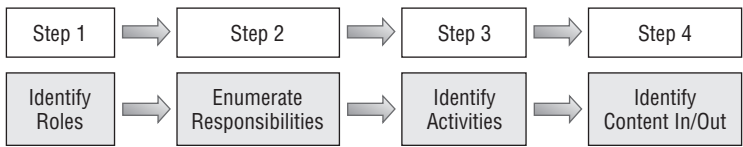| Step 1	Step 2	Step 3	Step 4
Identify Roles	Enumerate Responsibilities	Identify Activities	Identify Content In/Out

Figure 1-2: Role/activity modeling.

Here, we are making a few basic assumptions about the portal environment, such as the ability to capture metadata in the form of lists, the ability to control access to documents, the ability to build comprehensive workflows, and so on. Typically, what is needed prior to doing this kind of analysis is to build the necessary semantic tools to do the envisioning at a higher level. From the perspective of a business

analyst who may be new to the SharePoint environment, an essential ingredient is helping them to develop a realistic understanding of what features SharePoint provides out-of-the-box and what it does not, and then move on to a broader strategic vision that includes guidance for how those features can be applied to solving the particular business problem. In order to use the analysis as a foundation for building solutions, it is also necessary to identify reusable patterns that can be combined to reduce the overall complexity of a given problem by making it possible to decompose the problem into smaller, more manageable pieces.

I like to think of SharePoint in two dimensions: space and time. If you view it from the spatial dimensions, then the *share point* is the location in space at which you place information to be shared with others. It's like a glorified file share. It's a physical location for sharing information. Alternatively, if you look at it from the temporal dimension, then the *share point* is the point in time when collaboration becomes essential to your business process. In other words, there is a point in time when the complexity of working with shared content becomes so great that you have no viable alternative but to share the information. The solution requires more consciousness, so you have no choice but to collaborate. *SharePoint* is therefore a great name for this platform because it truly captures the spatial and temporal complexities that are inherent in the work we do and the activities we have been tasked with.

Consider the platform itself. We have a set of sites that support large numbers of users. We have a fixed topology that imposes a certain structure on the information stored within it. Perhaps there is a top-level site with departmental sites underneath it and end-users have the ability to create their own sites beneath those. With so many sites being created and so much information being stored in so many ways, the spatial constraint is to reduce the unnecessary duplication of information. But it's a temporal constraint as well. We want to enable users to find and reuse information as much as possible, just in the interest of time. Otherwise, they will fall into the same behavioral patterns they used when all they had were file shares. The reason there were so many duplicate copies of the same document was because someone couldn't find that first copy they created last year that was exactly on point for what they are doing now. Even if it was only 80 percent on point, they just don't have time to look for it.

Enumerating Roles

We need a way to determine what the workflow patterns look like because there are many ways to structure information to support those patterns. But there can be a real disconnect between the collaboration platform and what people in the organization are being tasked with. In many ways, collaboration presents a paradigm shift for the average information worker, who is used to working on a separate *island* of information. Now, there are other *nearby islands* they need to collaborate with, but they are still thinking in terms of *my island* and *your island*. They are not yet comfortable with the *shared repository* idea, especially when it potentially affects their ability to deliver work product.

The role/activity modeling exercise helps to clearly state and enumerate the roles that are involved. There are clearly defined responsibilities. That's an embedded relationship. You can't have the same responsibility in more than one role. By enumerating the list of responsibilities associated with each role, you have a way to identify the primary forces working against the collaboration initiative. When it comes down to it, people are motivated by their need to fulfill their perceived responsibilities. Adopting a new paradigm requires effort. It requires time to learn the new tools. It requires a commitment to learning. It is easier to persuade people to invest the additional time and energy if they think the new paradigm will better enable them to fulfill those responsibilities.

If you roll out a portal that does not facilitate the individual's ability to perform his or her assigned tasks, you are increasing the noise they have to deal with, and you are increasing the effort they must

expend. There is then little incentive for them to adopt the new paradigm. One of the strategies that Microsoft started with was to integrate the collaboration platform tightly with Office. This gives you the ability to build on knowledge they already have. You have the ability to *extend* rather than *shift* the paradigm. This works well for both the collaboration and aggregation activities.

Enumerating Responsibilities, Activities, and Tasks

The role/activity modeling exercise can really help to clarify and focus the structure of an information portal. Beyond that, it can also provide a foundation for building analytical models that can guide the construction of component libraries so they provide the essential tools required by users to meet specific business objectives. Figure 1-3 illustrates the relationships between the different parts of the model.
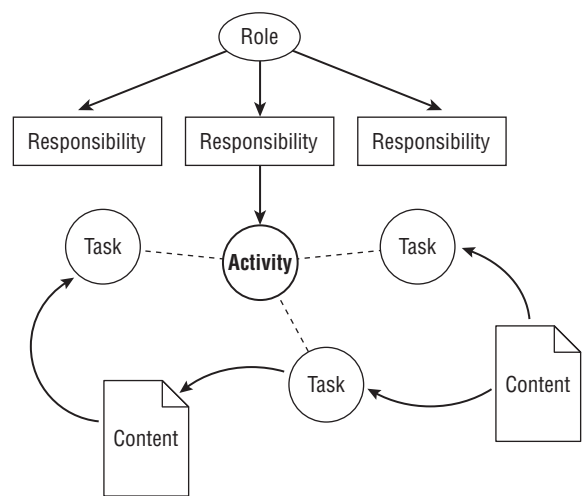
Figure 1-3: Role/activity modeling elements.

People don't really have a choice with regard to their responsibilities. It's part of their job descriptions. They must fulfill their responsibilities in order to satisfy their job requirements. The following table shows some common examples of roles and responsibilities.

Group	Role	Responsibilities
Administration	Administrative Assistant	Managing communications and scheduling activities to alleviate executive information workload
	Accountant	Collecting, auditing, and reviewing financial documents and related information
		Creating financial reports
	Bookkeeper	Collecting and distributing funds and issuing receipts
		Maintaining financial records

Group	Role	Responsibilities
	Operations	Ensuring that all operations are running smoothly
		Troubleshooting problems as they arise
Compliance	Comptroller	Ensuring that legal documents, contracts, and releases are properly signed, received, and filed
	Compliance Officer	Ensuring that the organization acts in accordance with governmental rules and laws as well as internal policies and guidelines
		Ensuring that governmental reporting requirements are met
Media/PR	Outreach	Contacting and maintaining relationships with the press
	Marketing	Identifying advertising opportunities
		Creating marketing plans
		Developing PR campaigns
		Interfacing with graphic designers
		Developing branding requirements and specifications
	Press Coordinator	Developing and approving talking points for media spokespeople
		Coordinating with the media outreach specialist
		Developing and maintaining lists of media spokespeople
		Developing and approving press materials
		Developing and maintaining a database of media contacts
		Interfacing with the publications group to ensure that materials are created
Human Resources	HR Manager	Identifying the personnel needs of the organization
		Recruiting, hiring, and managing the staff
	Mediator/ Counselor	Reviewing trends to identify potential problems
		Pulling together the appropriate resources to help people deal with difficult situations
		Creating a mechanism to enable people to find the support they need
		Acting as a mediator/facilitator
	Team Builder	Identifying needs, creating teams, and inspiring people to join them
		Creating and designing activities to build team identity and cohesiveness and sustainability within the team

Continued

Group	Role	Responsibilities
	Skills Manager	Creating and disseminating training programs and materials to keep staff and teachers up-to-date with new policies, programs, and legal requirements
		Identifying gaps in skills and developing tools for updating individual skills and ensuring that mandated policies are effectively applied

Getting people to simply articulate their understanding of what their responsibilities are can be helpful in itself. You can organize them easily in a simple spreadsheet with columns for Role, Responsibilities, Tasks/Activities, and Inputs and Outputs. You don't have to organize them into an elaborate model — just simply enumerate them. Later on you can project them onto a site topology that identifies the most appropriate tools that support those activities. They don't have to be connected into a workflow, either, although the ultimate tool would be one that ties the inputs and outputs to other activities within the same business process. That would yield a modeling platform that provides everything needed to map the process all the way down to the individual content elements flowing into and out of the system. Until we have such a tool, just getting users to articulate their understanding of the roles and responsibilities can be valuable.

A Simple Example

As an example, consider the role of compliance officer in an organization. The compliance officer is responsible for ensuring that the system is properly enforcing the core requirements of any records management system, namely, confidentiality, auditability, high availability, and so on. Figure 1-4 illustrates the technique applied to this role.

These are unique responsibilities, because the same responsibility cannot be associated with more than one role. Also, high availability in the context of records management does not mean up time, *which might be the responsibility of an operations staff or system administrators. It means ensuring that the information needed to manage a given set of records is always available and that the ability to comply with regulatory requirements is not compromised by day-to-day interactions with the affected documents.*
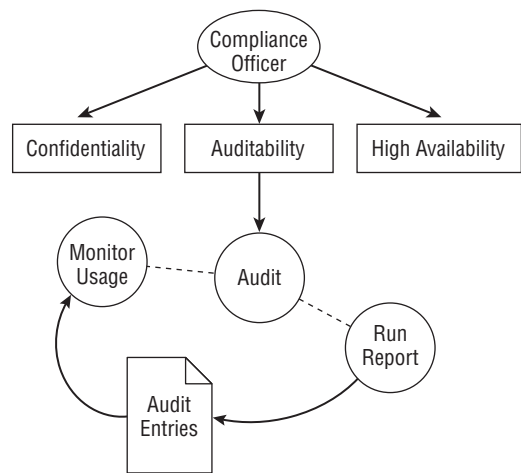

Figure 1-4: Role/activity modeling example.

Within the *auditability* responsibility, we want to derive a list of tasks that are essential to fulfilling that responsibility. When we are done, we want to be able to say, "If I perform these tasks, then the responsibility is fulfilled." So the tasks must be tied directly to the fulfillment. Within each task, there may be several activities that are part of the task. This just gives us a way to capture iterative processes that taken together make up the set of actions that must be performed. Keep in mind that each activity may feed other tasks. They may be ordered or unordered — it depends on the task.

Within each activity, there is information that is essential to performing the activity. Without that information, the activity cannot be done. We need to enumerate those pieces of information as well. Similarly, as a result of performing an activity, there is new information that is produced. We can say that every activity must produce new information or it is not an activity we are interested in modeling. However, metadata can also fulfill this requirement, so the very fact that an activity was initiated can be information that is produced as a by-product of performing the activity.

> *Typically, list items, documents, or e-mails are produced or consumed by the activities when modeling human interactions. The same technique can be applied to system level workflows; however, different kinds of information may be produced or consumed — perhaps more fine-grained. For example, when dealing with business-to-business workflows, we may be talking about packets of information in the form of WCF messages instead of actual documents.*

After going through this process, you can now build a site topology that presents all of the tools and information needed for a records manager or project lead to fulfill the responsibilities to which they've been assigned. You could provide a *Records Manager's Workspace* that includes links for performing a specific set of tasks. You could go further and limit the available tasks to only those that have been enumerated during the role/activity modeling exercise. You could also think about the web parts or custom Web Services needed to support those activities, whether they are presented sequentially or in random order.

For any given model, the level of granularity needed when defining the roles can vary. They can be specific to an industry, especially in the context of regulatory compliance that often involves roles whose responsibilities are driven directly by constraints imposed by a particular piece of legislation. One approach is to actually define the roles as off-shoots of each regulation. So, for example, we could define roles such as *HIPAA Compliance Officer* and *SOX Compliance Officer* and model the solution based on the specific regulatory issues faced by those users. Such an analysis would naturally lead to HIPAA-specific web parts that simplify searching for records that are affected by HIPAA rules, or a SOX-specific file plan that captures SOX-related metadata that, in turn, is used to set up the Records Center site to work with those records, and so on. Figure 1-5 depicts what this might look like.

> *The end-users involved in a records management application are not users of the Records Center site, but users of the collaboration portal —* collaborative *end-users. These are the people who are interacting with documents daily. These documents may or may not be promoted to* official record *status. In the simplest case, there are scenarios wherein the end-user selects a document and sends it manually to the Records Repository. Other scenarios involving end-users might require specialized tools beyond the simple* send to repository *functionality provided by the SharePoint UI. As an example, you might want to enable users to send many documents to the Records Center at once, and consequently may need to provide tools that allow them to specify all of the metadata for those records in a single batch.*
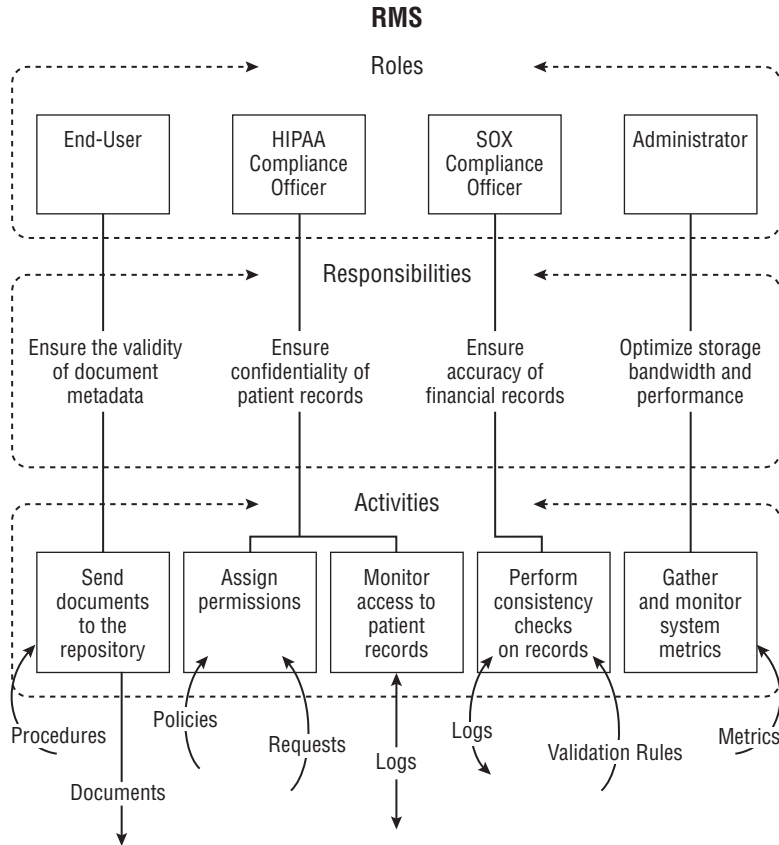
RMS

Roles

| End-User | HIPAA Compliance Officer | SOX Compliance Officer | Administrator |

Responsibilities

Ensure the validity of document metadata | Ensure confidentiality of patient records | Ensure accuracy of financial records | Optimize storage bandwidth and performance

Activities

| Send documents to the repository | Assign permissions | Monitor access to patient records | Perform consistency checks on records | Gather and monitor system metrics |

Procedures   Policies   Requests   Logs   Validation Rules   Metrics

Documents   Logs

Figure 1-5: RMS role activity model.

Factors to Consider When Applying the Technique

There are several useful factors to consider when using role/activity modeling:

❑ The discovery process empowers knowledge workers.

❑ Models can be created quickly using familiar tools, making the modeling exercise approachable by business analysts, developers, and end-users.

❑ Modeling artifacts (such as spreadsheets and Visio diagrams) directly support software component building and can help clarify and drive other development activities.

Discovery Empowers Knowledge Workers

Just by going through this discovery process, knowledge workers may suddenly understand what their responsibilities are, whereas before the exercise, their understanding of this might have been vague. It forces everyone to be on the same page in terms of identifying what the requirements are for a particular set of content elements and responsibilities. Another thing to observe is that it's very approachable both by business analysts and by those who are not business analysts, but who know what their

business responsibilities are. By being engaged in the role/activity modeling exercise early in the solution development cycle, knowledge workers can gain deeper insights into their business function and understand their existing responsibilities in new ways that can improve their performance. Common understanding forces everyone onto the same page, enabling them to share responsibilities more effectively and work better as a coordinated team, lessening the perceived need to take ownership of parts of the process that are beyond their immediate control.

Roles and Activities Are Easily Understood

In the case of one large volunteer organization (with more than 10,000 members), we pulled together people from IT, people from management, and people from the board of directors and the oversight group, as well as people who were being assigned these tasks, such as accounting and HR, with a wide range of skills and experience, but everyone in the room was comfortable with Excel. No one pushed back on the process. This enabled us to reach consensus quickly and figure out exactly what they needed in this particular portal environment.

Without a methodology like this, it all falls on IT to figure out what everyone needs. Experience in several different projects has shown that there can be a substantial time lag between when IT comes up with a design and when they present it to those who are going to use the SharePoint portal in their day-to-day activities. This increases the chance for failure if there was never any clear enumeration of roles and responsibilities. Applying this methodology can really help with user adoption of any kind of portal design.

Content modeling and data validation are interrelated. When building a content model, there is a natural progression from the role/activity modeling exercise to the construction of a more detailed XML schema that describes the content and its interactions with various business processes. From that point onward, an API can be developed gradually using generated classes that are extended with domain-specific interfaces as they are discovered.

Modeling Artifacts Support Component Building

After the role/activity modeling exercise is complete, you will end up with some number of spreadsheets and diagrams that describe the model elements and how they are related. We can call these *artifacts* of the modeling exercise. What's really interesting is that by defining these artifacts in a structured way, we now have tools that we can use to go ahead and begin declaring our content types. The modeling exercise has already identified the major content types as well as their interdependencies. What is left is the actual fields that make up each type. The field derivation is fairly straightforward at this point. Just examining the tasks and looking at what they consume and produce is enough to identify existing site columns and field types or to design new ones. Finally, the context provided by the roles and responsibilities enables us to create separate schemas that describe each type in context as it moves from one phase of its life cycle to the next.

This means that you can build XML schemas to support various layers of the solution. As an example, consider a meeting agenda for a board of directors. This information is required by the board secretary, but is also required by the individual board members. So the meeting agenda has a life cycle that interacts with both roles. Therefore, when defining the metadata for the meeting agenda, you can focus on the metadata that is specific to the board secretary and consider that metadata separately from the same meeting agenda when it is being used by a board member. Likewise for any of the other items, you can easily see when a particular content element is being referenced or consumed by multiple roles.

As we apply the role/activity modeling technique to records management, you will see that this ability to capture high-level details of the content life cycle in a schema that maps each phase to a discrete set of roles and activities makes the XML schema an important tool for building any kind of content management solution. In the next section, we'll examine what are the most important elements of such a schema that could be used specifically for managing official records.

Developing a File Plan

At the heart of any records management system is the *file plan.* Records managers and compliance officers are accustomed to creating file planning worksheets that describe the kinds of documents that their organization will treat as *official records*. The *file plan* describes where each type of record should be stored, how long it should be kept, and the manner and conditions under which it will be archived or destroyed. A traditional file plan may also include additional information that is used to categorize documents and to assign tasks to the persons responsible for managing each record type.

The fundamental concept of an *official record* is intended to convey the notion that at some point in its life cycle, a document may serve as evidence of some transaction that has taken place within the organization. For instance, when a contract or legal agreement is signed, then the contract itself serves as evidence of the agreement. It's not the *only* evidence of the agreement, but taken together with other evidence, it can serve to clarify the intentions of both parties. Therefore, it should be possible to somehow *freeze* the document in its current state such that a snapshot of the document is stored in the system so that it can be retrieved and reviewed later.

> *For certain scenarios, taking an actual snapshot of the document may suffice. However, storing only an image of the document is not as useful as keeping all of its metadata, macros, embedded objects, and so on.*

The following table lists the basic elements of a file plan, as defined by Microsoft. They provide a starting point for analyzing documents and describing the types of records needed to manage them. The resulting worksheet can then be used to design an effective MOSS Record Repository.

Element	Description
Record Type	This is the name of the record type and typically matches the name of the content type associated with the document.
Description	This is a brief description of the record type that should be targeted at content managers so they understand the rationale for "promoting" the document to an *official record*.
Media	This describes the format in which the record is stored, such as MP3, HTML, Word 2007, and so on.
Category	This is a general categorization that can be used to group similar record types together, for example, when deciding which document libraries will house submitted documents.
Retention	This is a statement of the required retention period for records of this type. The retention statement will ultimately be used to determine how the expiration policy for the document is configured. It should therefore include all of the information needed to perform that configuration.

Element	Description
Disposition	This is a description of what will happen to the document when its designated retention period ends. This will be used to select the appropriate expiration action for the document and will therefore also influence the configuration of the expiration policy applied to the document.
Contact	This is the name of the compliance officer or content administrator assigned to documents of this type.

Typically, this information is captured in a spreadsheet and then is referred to by all members of the compliance team, which may include lawyers, business analysts, and IT personnel. This spreadsheet can become a useful artifact because it can be easily uploaded to a document library, revised, approved, and then used to drive the manual construction and maintenance of the Records Repository. From this one document, all of the required content types can be identified and linked to the appropriate routing types. Custom routers can also be built and installed if necessary, and document retention policies as well as other information policies can be configured and tested.

Ultimately, we'd like to move beyond the manual model represented by the *static* file plan described above toward a more automated approach, wherein a *dynamic file plan* is used to drive the process of adding the required components into an existing Records Repository. An automated or semi-automated approach would fit in well with the day-to-day operation of a typical Records Center by enabling compliance officers and content managers to deal more effectively with constantly changing requirements and regulations. Ideally, we'd like to publish the file plan using a set of SharePoint features. But before we can delve into the mechanics of building dynamic file plans using the MOSS Records Center API, we first need to cover the manual steps that are required to set up a Records Repository using the out-of-the-box Records Center site template.

Identifying Roles and Responsibilities

The process of associating roles and responsibilities with the file plan flows very naturally from the role/activity modeling exercise. We simply copy the roles into the file plan, describing each one using information that we've captured in the spreadsheet. We can follow a similar process when enumerating the responsibilities, keeping in mind our rule that there can be only one role associated with each responsibility. By enumerating the roles within the file plan, we can more easily see how different aspects of the plan are affected by and serve the various roles.

In this way, we can think of file planning as an extension of the role/activity modeling exercise, the key difference being that whereas file planning is content-centric, role/activity modeling is more process-driven. This translation from process- to content-centric views while creating the file plan is key to developing a clear understanding of how a particular kind of content drives the business process. The role/activity modeling exercise then becomes an essential preliminary step so that we thoroughly understand the process before attempting to refine our understanding of each content element that feeds it.

Identifying Applicable Policies and Procedures

To develop a comprehensive file plan that can support different operations at various stages of the overall processing life cycle for a record, it is important to identify the policies and procedures that govern what happens to the record at each stage. One approach might be to define the policy very simply as

just a descriptive text statement attached to each record. That text could then be presented to the content manager or to other persons involved in the document processing sequence, and could explain the purpose of the policy and the reasons for its inclusion, perhaps tying it back to a particular piece of legislation or to a particular organizational rule or procedure.

Another approach might be to attempt a more granular description of the policy by breaking it down into its constituent parts. Using the expressive power of XML, for instance, we could begin with a Policy element and then identify different parts of the policy as nested subelements. Those subelements might refer to other elements in the file plan, such as the roles and responsibilities most affected by the policy, and so on.

Identifying Custom Routing and Workflow Requirements

As we develop a deeper understanding of how a particular content element feeds the business process, we're really starting to talk about workflow. We're looking specifically at how a particular content element flows out of the collaborative environment and into the more structured and restrictive environment of the Records Repository. We are basically using the file plan to define a path for the record to follow, and as part of that process we are examining what happens to the record as it moves along that path.

This is where custom routing requirements may begin to emerge. If, for example, we determine that the primary requirement for handling a given set of documents is to keep track of what happens to them, then we may need enhanced tracking that is more detailed or comprehensive than the tracking mechanisms that are provided out-of-the-box. For other record types, such as patient records, the dominant requirement might be confidentiality, in which case, we may need a special router that attaches more restrictive permissions to the record when it is stored.

Identifying Document Categories and Groups

Categorization is a natural part of any content processing mechanism. We do this automatically whenever we decide what name to give a folder that will contain new documents that we create. We do the best we can to identify the proper folder, and we typically name the folders based on our current understanding of the document's intended purpose. Later, the primary purpose of the document may change, and we often end up copying the document to a different folder and renaming it.

When building a file plan, it is important to capture as much information as possible about the groups a given record might belong to as well as the categories it might be associated with. This supports the construction of different views of the record while it is being processed. You can also use the categorization information to define queries for retrieving and manipulating only those documents you are interested in.

Identifying Document Sources

The process of finding documents that match the file plan can be highly subjective. Describing the sources of those documents in an objective way can help to simplify the process and can also provide a foundation for building expertise around the file plan that is not dependent on any one person. One way to do this is to include information that can be used to create a query for finding the documents to which the plan applies.

This is technically more of a document management than a records management activity, because it applies to *active* documents that have not yet moved into the records management process. However,

capturing the relevant metadata, location, keywords, and other properties that determine whether the plan applies to a given document is an important part of the planning process, and the file plan is the most convenient location for storing this information.

Analyzing Storage Requirements

Storage information can be included in the file plan to support capacity planning for the Records Repository and is related to the identification of document sources. Using a query to identify and locate record candidates would allow you also to calculate the probable size of the Repository before actually constructing it. This information might also influence how the Repository is organized.

Analyzing Security Requirements

Security and access constraints should also be included in the file plan so that permissions can be set up automatically for certain documents. For example, if you're developing a file plan for healthcare documents that are governed by the Health Insurance Portability and Accountability Act (HIPAA), then confidentiality rules may apply such that when the records are processed, only certain people are allowed to see them. Having a description of these rules in the file plan enables any processing components to assign the required permissions to the appropriate people or groups.

Another aspect of security is the identification of the persons who will be responsible for the management of the record once it moves into the repository. Thus, there may be multiple levels of security that must be described. One approach is to identify a *records manager* with a certain set of permissions and then to add a separate list of roles and their permissions.

Summary

Records management is driven by regulatory compliance, which imposes core constraints on any records management system. These include confidentiality, information integrity, high availability, adherence to well-established policies, and auditability. The MOSS Records Management infrastructure provides tools that support each of these requirements.

This chapter introduced the concept of *content modeling*, which helps to develop more effective records management solutions by identifying the key content elements that drive a business process. The role/activity modeling technique was introduced as a straightforward methodology that identifies the primary roles and responsibilities involved in a business process and then maps them to the tasks and content elements needed to fulfill them. This approach is easily understandable to business analysts as well as to knowledge workers and IT staff, which makes it an ideal candidate for bridging the gap between the groups involved in the development of business process automation solutions. The role/activity modeling technique can be applied easily using readily available tools like Excel and Visio and offers a way to ensure consistent binding of metadata to content elements, the people who interact with them, and the business processes that produce and consume them.

At the heart of any records management strategy is the *file plan*, which is a document that describes the criteria that determines which documents shall be treated as *official records* and how they should be processed. This chapter enumerated the essential parts of an effective file plan and showed how the file plan can provide a strong foundation for further development activities.