# Sitecore CMS

# and Microsoft SharePoint 2007

Author: Rolf Kraus, Sitecore USA Inc.

Date: September 2006

Release: Revision B

Language: English



**Sitecore Corporation**
Meldahlsgade 5, 4.
1613 Copenhagen V
Denmark

Sitecore.
Content Delivery

www.sitecore.net
info@sitecore.net

T: +45 70 23 66 60
F: +45 70 23 66 70

Contents:

# 1 Purpose

This document was developed in response to questions that Sitecore has received from potential customers, industry analysts and press who want to understand the differences and synergies between Sitecore CMS and SharePoint 2007.

With the limited information that is currently available about SharePoint 2007 (July 2006) this document will attempt to provide insight into answering many of the common questions that Sitecore has received:

- "I understand SharePoint 2007 contains CMS functionality…why should I still consider Sitecore CMS for web content management?"
- "How does Sitecore CMS functionality compare with that of SharePoint 2007?"
- "Do I need both Sitecore CMS and SharePoint 2007, or just one of the products?"

This document clarifies how both Sitecore CMS and SharePoint 2007 each have a specific purpose and use within an organization. This clarification should lead potential customers to consider one of three buying decisions that involve one of the products or both:

- Purchase Sitecore CMS and SharePoint 2007
- Purchase Sitecore CMS and not SharePoint 2007
- Purchase SharePoint 2007 and not Sitecore CMS

## 1.1 Summary Analysis

The following summarizes key web content management considerations while outlining key evaluation factors for customers and analysts to consider.

- SharePoint 2007's web content management solutions is a force fit of the Microsoft CMS 2002 functionality into the basic "List and Library" architecture carried over from Microsoft SharePoint Server 2003. Microsoft has based a significant product with broad functionality on cumbersome legacy architecture.
- SharePoint 2007 does not separate content from presentation. Each Page Template contains controls that only render content from the content List the template is bound to—limiting the flexibility of page templates. In addition, to render content in a different language, on a different device (i.e. PDA, Printer-friendly, RSS, etc.) or with different branding, requires separately managed copy of the original site to be created. The number of sites to manage can balloon quickly based on the permutations of brands, languages and other factors.
- Microsoft favors proprietary, rather than standards-based technologies, for many common content and presentation mechanisms and constructs. This will force web designers and developers to either live with the limitations of packaged controls and interfaces or struggle to learn a proprietary APIs to deliver common web-based functionality. This will either increase the delivery costs of solutions, or limit solutions to the "out-of-box" functionality included with SharePoint.
- SharePoint 2007 has an overly complicated approach to modeling content types and defining relationships for website content and assets. The constructs of Site Content Types and Site Columns together define content behavior, attributes and types. Intended to provide extreme flexibility when modeling enterprise content, the constructs used to define web content require a significant effort in planning prior to implementation, while limiting the ability to inherit and propagate future changes to content.

Like many ECM products, SharePoint was not designed around web content management, but rather an architecture supporting collaboration, document management and ad-hoc Intranet portals. SharePoint is a generalized content management architecture that being applied to the

web content management problem. This approach results in a product that is not optimized for web content management, forcing organizations to invest more resources learning, implementing and working around the product to deliver basic solutions.

## 1.2 SharePoint 2007 and Sitecore CMS Work Together

Sitecore CMS has an optional SharePoint Connector module that provides the ability to copy content from or synchronize content with SharePoint. The SharePoint Connector module works with both structured content managed in SharePoint lists and document-based content stored in SharePoint libraries.

When copying content from SharePoint, the module will literally duplicate the structured content or documents within the Sitecore repository. When synchronizing content, the module will reference the structured content or document within SharePoint's repository.

In either situation, whether the content is copied or synchronized, SharePoint structured content and documents will appear and behave as native Sitecore content. This enables content editors, web administrators and web developers to apply Sitecore's tool-user security, extranet security, workflow, versioning and other attributes to this content. In addition, SharePoint-based content can now be web-enabled through Sitecore—allowing the quick and simple delivery of SharePoint content to any Sitecore managed web site.

Using the module, web administrators can map SharePoint lists to Sitecore structured content items and SharePoint document libraries to folders within the Sitecore Media Library. Once the mapping is complete, Sitecore will automatically copy or reference the specified content within SharePoint.

## 1.3 When to Choose SharePoint 2007 and Sitecore CMS

Organizations considering SharePoint and Sitecore will want to understand the strengths and limitations of each product. SharePoint is clearly focused on the enterprise content management (ECM) and ad-hoc internal portals, while Sitecore has focused on web content management (WCM). With each product focusing on different problems, the products complement each other to deliver strong content management solutions for customers.

### 1.3.1 When to Choose Sitecore CMS and SharePoint 2007

Choose to purchase and implement Sitecore CMS and SharePoint 2007 when organizations require both a strong web content management and document management platform. Sitecore CMS is a strong platform for web content management and delivery for web sites and portals. By enabling organizations to deliver web-based functionality more quickly, with less cost and with greater on-going flexibility than SharePoint, Sitecore should be considered as the web platform of choice.

With Sitecore's strength in web content management and delivery, SharePoint provides a strong complement with its document management abilities. Organizations that value the ability to management documents, create ad-hoc intranet portals and tightly integrate these activities with Microsoft Office should consider SharePoint.

The ability to leverage the strengths of both Sitecore and SharePoint together is easily done through the use of the Sitecore SharePoint Connector module. This module will enable organizations to deliver any content contained in SharePoint to the web, taking full advantage of Sitecore's management and delivery capabilities.

### 1.3.2 When to Choose Sitecore CMS and Not SharePoint 2007

Choose to purchase and implement Sitecore CMS and not SharePoint when organizations have limited document management or internal portal needs. When organizations are predominately focused on managing and delivering content to public websites, extranet and portals, then Sitecore should be considered as the web platform of choice.

Sitecore not only provides a solid web platform, but also has the ability to store and manage document-based assets that may be used in a web site. Sitecore provides the ability store, version and workflow document-based assets within its own repository before delivering them to the web to provide a single platform for web and document management.

### 1.3.3 When to Choose SharePoint 2007 and Not Sitecore CMS

Choose to purchase and implement SharePoint and not Sitecore when organizations have strong asset management and internal portal requirements, while having limited requirements for public websites, extranet and portals.

When organizations want to quickly create project or department-specific portals for managing projects, documents and lists of structured content, then SharePoint should be considered the portal and asset management platform of choice.

# 2 Products

Microsoft's introduction of Microsoft Office SharePoint 2007 (SharePoint) as the successor to both SharePoint Portal Server 2003 and Content Management Server (CMS) 2002 has prompted both excitement and questions about the platform's functionality. This section lays the ground work for understanding SharePoint 2007's in comparison Sitecore CMS (Sitecore).

## 2.1 What is SharePoint 2007?

SharePoint is a suite of various server functionalities that Microsoft positions as an Enterprise Content Management (ECM) platform. Historically, organizations have used SharePoint 2003 as a collaborative environment to capture, store and manage lists of structured content and documents. SharePoint 2007 expands the type and quantity of content that may be managed, while providing more flexibility in how to deliver the content to both internal and external consumers.

SharePoint now provides rich ECM support for more content types, including documents, web content, records, forms-based data and enterprise data. In addition, SharePoint provides better ways to manage, find and distribute content through its strong enterprise search, workflow and portal frameworks.

SharePoint provides knowledge workers with a unified approach to managing a wide variety of structured and unstructured content from within a single portal environment. Its purpose is to enable people to quickly create and deploy ad-hoc environments to manage and distribute information in a flexible manner.

Please reference sections 2.8.1 and 2.8.2 to learn about new functionality within SharePoint 2007.

## 2.2 What is Sitecore CMS?

Sitecore CMS is a web content management system (WCMS) and portal platform. Sitecore enables organizations to build, deploy and manage web-based business solutions, such as web applications, public websites and secure portals which require enterprise-class WCMS functionality, integration options and scalability.

Sitecore is built upon the Microsoft .NET Framework to provide organizations with a platform that seamlessly integrates Microsoft servers and .NET applications. In addition, Sitecore embraces standards-based (W3C) technologies to provide a contemporary, best-practices approach to content management.

Sitecore's architecture and approach is optimized for the management and delivery of web content. This optimization enables organizations to build and manage web-based solutions with less time and effort than competing platforms, thereby reducing the total cost of ownership (TCO) for both Sitecore and the resulting business solution.

## 2.3 Market Positioning

### 2.3.1 SharePoint 2007

Clearly SharePoint is positioned as the single ECM platform from which knowledge workers can access, manage and interact with virtually all of an organization's internal data and content. SharePoint's tight integration with the Microsoft Office environment and its ability to enable the quick creation of basic ad-hoc portals and websites positions it as a strong departmental and inter-departmental solution.

Microsoft clearly wants to "kill two birds with one stone" with this product. With the significant functional limitations of the previous SharePoint Portal Server 2003 and the sun-setting of Microsoft Content Management Server (MS-CMS) 2002, Microsoft took SharePoint 2007 as an opportunity to combine the functionality of both products into one.

This combination has forced web content management, enterprise content management and Intranet portal functionality into a common approach, fundamentally based on the legacy SharePoint architecture, which Microsoft now positions as the singular “do all” platform.

SharePoint’s broad promises, particularly by positioning it as Microsoft’s premier web content management platform, will be reviewed and tested by many organizations employing Microsoft technologies that are looking to replace their aging MS-CMS installations or legacy CMS products not based on the .NET framework. Sitecore believes many of these organizations will evaluate alternatives to SharePoint for web content management functionality.

### 2.3.2 Sitecore CMS

Sitecore CMS is positioned as the premier Microsoft .NET-based web content management and portal platform for medium to large organizations requiring enterprise-class functionality, integration and scalability. With Sitecore’s contemporary, open architecture, it is well-received by organizations looking for a strategic platform with which to develop complex web-based applications and portals that must integrate tightly into an existing enterprise environment.

Sitecore limits its focus to web content management with clear examples of integration to a variety of enterprise content repositories and applications, including document management systems, ERP, CRM and other solutions. Sitecore provides a platform that can integrate with any content source and provide a consistent management experience for this external content.

As a platform optimized for development and management of web-based solutions, Sitecore provides organizations a complement to existing ECM solutions that can deliver more business value with less resources than with general-purpose ECM solutions when developing web-based business solutions.

## 2.4 Web Content Management Approach

SharePoint 2007 and Sitecore approach web content management very differently. These differences are evident in how both content management systems (CMS) address four components to deliver a web site:

- Content definition, management and storage
- Page definition and rendering
- Website definition
- Publishing and delivery

The following sections will provide a brief review of the four components for both SharePoint 2007 and Sitecore CMS.

### 2.4.1 SharePoint 2007

#### Content Definition, Management and Storage

SharePoint 2007 shares its approach with its predecessor SharePoint Portal Server 2003 by storing all content in Lists and Libraries. Lists are based on one or more Site Content Types containing Site Columns.

- Site Columns are metadata fields (all fields in SharePoint are called Metadata). Nineteen Site Column types are provided (i.e. Text, Currency, Yes/No, etc.).
- Site Content Types are a reusable collection of properties for a category of content. Content Type properties include references to metadata (defined by Site Columns), workflows, document templates, custom forms and custom policies. Site Content Types may inherit other Content Types, thereby create compound Content Types.

- Both Site Content Types and Site Columns are created at the site level and are templates. The Site Content Type and Site Column templates can be copied to a List to create customizable instances called List Content Types or List Columns.
- Lists manage Documents or content items, like addresses or news articles which, are defined by a list type as a collection of fields known as columns.
- Libraries manage file-based assets such as JPEG or a PDF files.
- Both Lists and Libraries provide versioning, workflow, alerts and check-in/check-out functionality.

All Lists and Libraries are stored in a proprietary format within Microsoft SQL Server.

### Page Definition and Rendering

SharePoint 2007 defines web pages through Master Pages and Page Layouts. Master Pages are new to SharePoint 2007 and did not exist in MS-CMS 2002 as they are part of ASP.NET 2.0.

- Master Pages are templates that contain common website elements such as menus and footers which are shared across many pages.
- Page Templates are templates that contain controls and web parts that render content on the web page. Each Page Template is associated with a specific list and can only render content from that list.
- A SharePoint web page is the pairing of a Master Page, which renders common page elements, and a Page Template, which renders list contents and web parts.

Content editors can create mix-and-match Page Templates and Master Pages to create their own web page combinations.

### Website Definition

SharePoint 2007 uses the same mechanisms for internal portals and public websites, though toolsets and requirements for each can vary significantly. All sites are defined by a Top-level Site and its sub-site children created within a Site Collection.

- The Site Collection is literally a collection of sites with single parent, the Top-level Site, which defines an owner and administration settings.
- The Top-level Site is the parent, or root, site for any number of sub-sites created within the Site Collection. It contains a document library called the Master Page and Page Layout Gallery that contains Master Pages and Page Layouts templates only visible from within this Site Collection.
- Any combination of sub-sites, pages and directories can be created beneath the Top-level Site.

SharePoint supports any number of independent Site Collections per server.

### Publishing and Delivery

SharePoint 2007 provides web page delivery dynamically using ASP.NET to assemble and render the templates, controls and associated content. SharePoint performs the task of moving content, or publishing content, to a production web server with the Content Deployment feature.

- Content Deployment moves content from one Site Collection to another, either on the same physical server or out external servers. Content Deployment uses Paths and Jobs to define and perform the publishing tasks.
- Paths define the relationship between the source and destination Site Collection. The Path contains information about what application and Site Collection is being deployed and authentication information for the destination server.

- Jobs are associated with paths and determine which sites in the source Site Collection will be deployed and on what schedule.

Each destination server is a SharePoint 2007 server.

### 2.4.2 Sitecore CMS

#### Content Definition, Management and Storage

Sitecore manages and stores both structured content and file-based content.

- Structured content is defined by a Template which represents a specific content type, such as a news article or job posting. Templates are a collection of content fields and inherited attributes. Template attributes include CMS user and Extranet security, default workflow and other aspects which affect the behavior of content items of that type.
- Templates reference multiple Layouts (discussed in detail below) which are inherited by content items. Each Layout defines how to render the content item for a specific device such as a PDA, RSS or browser. Content items are automatically rendered correctly for the device that is requesting the content item.
- Structured content items, based on Templates, are stored as Extensible Markup Language (XML) within Sitecore's repository. Parent-child relationships between content items form a tree which represents the information architecture (IA) of the website.
- Structured content items contain the content for each defined language. The default language, set programmatically, is automatically rendered for all content items within the website.
- File-based content, or Media Items, are uploaded and stored within the Media Library. Each media item inherits metadata based on its media type (i.e. JPEG, GIF or PDF). Metadata for each item in the media library is stored and managed as structured content. Media items themselves are stored as blobs within a database and are cached on disk when requested.
- Management of content items, both structured and file-based, is done within content browsers that provide a visual representation of the relationships between all content items.

Sitecore content is stored within a database. Sitecore supports Microsoft SQL Server, Oracle and MySQL. Sitecore enables multi-database support through its Data Storage Abstraction layer.

#### Page Definition and Rendering

Sitecore defines web pages with ASP.NET-based Placeholders, Layouts, Sublayouts and Renderings as wells as .NET and XSL content renderings. All presentation components have access to the entire XML content repository.

- Placeholders define a region of a Layout or Sublayout that can be populated with different Sublayouts and/or Renderings. Unlike the static Master Page and Template relationship within SharePoint, Placeholders are populated dynamically at runtime, based on design-time definitions, business policies or personalization preferences.
- Layouts are page templates that provide page structure and are populated with website elements, or Renderings, that are common across many pages. Layouts are based on ASP.NET web page (.ASPX) files.
- Sublayouts are page templates that provide page structure and are populated with Renderings or Placeholders. Sublayouts may also be used as reusable transactional forms (i.e. Login form, Database look-up form). Sublayouts are based ASP.NET web user controls (.ASCX) files.

- Renderings are Extensible Stylesheet Language Transformations (XSLTs), ASP.NET controls or Web Parts that generates an element of the page (i.e. News article, Menu, Breadcrumb, etc.). Most Renderings access Sitecore's XML repository and Media Library from which they render content.

A page definition, the combination of a Layout with nested Placeholders, Sublayouts and Renderings, is associated with each Template, or content type. When a content item is requested from Sitecore, the items associated page definition, based on its Template, determines how the item is rendered in the context of a web page.

### Website Definition

Sitecore defines any website, an Intranet, Extranet or public facing site, by the content items within the XML repository.

- Parent-child relationships organize the content to form a tree of items within the repository. Sitecore supports multiple independent branches of the main content tree within the repository that form sub-trees which represent one or more websites.
- URLs within Sitecore do not correspond to files, but rather to content items within the XML repository. Domains may be mapped to any content item within a tree or sub-tree to identify the root path for a website. The URL path following the domain is the actual path to the content item within the XML content repository itself. Sitecore can support any number of websites with distinct domains and URLs from a single repository.
- The structure and relationship of the content items within a tree define the information architecture of a website. This information architecture can be used to drive navigational elements, like the main navigation, left navigation and bread crumb trails, for example. Changes to the structure of the website automatically are reflected in the navigational elements.

Sitecore can manage any combination of sites and sub-sites from a single repository, providing a unified administration environment for all websites.

### Publishing and Delivery

Sitecore provides web page delivery dynamically using ASP.NET to assemble and render the Layouts, Sublayouts and content Renderings on request. Sitecore performs the task of moving content, or publishing content, on the local CMS server and out to any number servers in one or more content delivery web farms.

- Sitecore leverages ASP.NET and Internet Information Servers (IIS) to dynamically assemble and build web pages when requested. Sitecore looks at the context of the received request, which includes the URL, the requesting device (i.e. PDA, Printer-friendly page, Browser, etc.) and requesting person (i.e. Anonymous, Partner, Customer, etc.), to determine which content item to render, what web page definition to use and what content trim and secure.
- Sitecore contains multiple databases, two of which are the staging (work-in-progress) and production content databases. Publishing within a single server environment moves content from the staged content database to the production content database.
- Sitecore supports multi-server environments by moving assets from the Sitecore CMS server to Sitecore Web Farm servers using the Staging Module. The Staging Module moves content, security policies, media items and file-based assets (i.e. DLLs, .ASPXs, etc.). The Staging Module uses database replication, SOAP and FTP to communicate between servers.
- Sitecore supports full, incremental and synchronized publishing models.

Sitecore supports the movement of content between any combination of Sitecore CMS and Sitecore Web Farm servers, including geographical distributed environments.

### 2.4.3 Points of Comparison

| Content Definition, Management and Storage | |
|---|---|
| **Sitecore CMS** | **SharePoint 2007** |
| W3C-standards based XML content repository for all structured content and metadata for media items.<br><br>Media items are stored as blobs within the database (of choice) and are referred to by media-specific metadata. | Proprietary content structure and storage based the notion of Lists and Libraries that is inherited from the SharePoint Portal Server 2003 legacy. |
| Native content storage is possible on MS SQL Server, MySQL and Oracle databases.<br><br>In addition, Sitecore ships with built-in managed-code transactional database for free. | Native content storage is limited to the MS SQL Server database. |
| Simple content definition:<br><br>A content item type, known as a Template contains fields. Templates have workflow, security and presentation definition attributes.<br><br>Templates may inherit other Templates to create compound Templates.<br><br>Fields have type, versioning, validation, language and security attributes.<br><br>New content item instances (XML content nodes) are based on the Template definition and inherit its attributes. | Complex content definition:<br><br>Site Columns define content fields and their type.<br><br>Site Content Types define workflows, document templates, custom forms and custom policies. Site Content Types reference one or more Site Columns. Different Content Types may reference the same Columns.<br><br>Site Content Types or Site Columns are copied to a List to become a List Content Type or List Column. A List may have multiple Content Types and Columns.<br><br>New content item instances are created within the List to inherit the attributes of the Content Types and Columns of the List. |
| New content instances (XML content nodes) may be created anywhere within the XML content tree (repository) or be limited by policies.<br><br>The logical relationship between content items (i.e. parent-child, kin or ancestor) is implied by their relationship to each other within the tree. | New content instances may only be created in Lists and Libraries.<br><br>There is no logical relationship between content items, other than a list-based relationship. |
| New content instances inherit all the attributes of the Template definition.<br><br>New or changed attributes are automatically inherited by existing content item instances from the Template definition. | Because Site Content Types or Site Columns are copied to a list, any changes to Site Content Type or Site Column attributes are not reflected in the List Content Types or List Columns.<br><br>These changes must be copied again to all Sites and Lists. |

| | |
|---|---|
| Content items understand how to render themselves in the context of a web page through reference to Layouts within a Template. | Content items have no knowledge of how render themselves in the context of website. Content is stored in a list (fundamentally the structure of flat file) with no knowledge of presentation. |
| Content items have Globally Unique Identifies (GUIDs).<br><br>Regardless of the name of a content item, it is still regarded as unique within the repository. | List items are unique based on their name, not on GUIDs. No two items in a list may share the same name.<br><br>For instance, SharePoint web pages are stored within a Page List with prevents any two web pages within a site to have the same name, since no two pages in the List can have the same. |
| **Page Definition and Rendering** | |
| **Sitecore CMS** | **SharePoint 2007** |
| Presentation is fully separated from content.<br><br>Layouts and Sublayouts can be populated with any number of different Renderings.<br><br>Given permission, renderings may access any content items in any site within the repository. | Presentation is not separated from content.<br><br>Page Templates are bound to specific content Lists or Libraries, and only able to render content from them.<br><br>The scope of Lists and Libraries are limited to the Site or sub-Site they are created in and therefore prevent Page Templates from rendering content in other Sites. |
| Flexible web page definition is performed by layering Layouts, Sublayouts and Renderings with the use of Placeholders.<br><br>Any depth and combination of Sublayouts and Renderings may be layered upon a Layout. | Limited web page definition is performed by defining two layers, the Master Page and Page Templates.<br><br>Page Template combinations are limited by the List they are bound to and the fact that Page Templates can only be a single layer. |
| Full reuse of all page elements.<br><br>Reuse of all Layouts, Sublayouts and content Renderings by any web site defined within the repository. | Limited scope of reuse of page elements.<br><br>Reuse of Master Pages within the Sites or Sub-sites they are created in. Reuse of Page Templates within the Sites or Sub-sites that contain the Lists they are bound to. |
| Sitecore Placeholder defines a region on Layout or Sublayout that is populated at runtime, either by a design time definition or programmatically, with Sublayouts and Renderings. | SharePoint has no equivalent mechanism, other then the single layer abstraction provided by Master Pages and Page Templates. |

| | |
|---|---|
| Placeholders provide tremendous flexibility in the runtime creation of a web page for page element reuse, real-time personalization and security. | |
| **Website Definition** | |
| **Sitecore CMS** | **SharePoint 2007** |
| Any number of web sites and sub-sites may be defined within the repository. Branches at any level can be cut-off to create new sites and content in branch may appear in multiple sites. | Any number of sub-sites may be defined Top-level Site within a Site Collection. A server can host any number of Site Collections. |
| Information architecture (IA) of website is defined by the relationships and structure of the content items within the repository. | Information architecture (IA) of a Site Collection is defined the relationship of the Top-level Site, its sub-Sites and pages (Master Page and Page Template). |
| Domains may be mapped to any content item within a tree or sub-tree to identify the root path for a website. | Domains may be mapped to the Top-level Site and its sub-Sites. |
| **Publishing and delivery** | |
| **Sitecore CMS** | **SharePoint 2007** |
| Sitecore leverages ASP.NET and Internet Information Servers (IIS) to dynamically assemble and render all layers of a web page on request. | SharePoint leverages ASP.NET and Internet Information Servers (IIS) to dynamically render web pages and controls on request. |
| Sitecore moves content, security policies, media items and file-based assets (i.e. DLLs, .ASPXs, etc.) via database replication, SOAP and FTP to communicate between servers. | SharePoint use Paths to define the relationship between the source and destination Site Collection, while Jobs determine the deployment schedule.<br>Deployment is performed via database replication. |

## 2.5 Other Web Content Management-related Functionality

### 2.5.1 Multi-language, Multi-device and Site Re-branding Support

**SharePoint 2007**

SharePoint uses site Variations to provide the ability to create parallel sites for support of alternate devices (i.e. PDA, Phone, and Braille-reader), languages and website branding. To create a site Variation, the administrator performs the following steps:

- Configure the Variation setting to create new .ASPX redirect file for the root of the site.
- Create Variation labels for the existing site and the new Variations. Variation labels are names associates with each Variation, such as “en-US”, “fr-FR” or “PDA”, that represent the intention of the variant site.
- Identify the source site, such as “en-US” for example.
- Create the Variation Hierarchy which creates and copies the source sites and pages for each Variation Label identified.

This process copies all the pages, lists and libraries to the Variant site as fully independent copy of the original.

Adding a new page to the source site will generate an Approval workflow. Upon approval the target pages are created in the Variation sites. Then each of the new pages within each Variation site can be translated or locally modified to support the primary intention of that site Variation.

## Sitecore CMS

Sitecore addresses the language, device and site re-branding support requirements with mechanisms that do not require the duplication of content or page definitions. Sitecore's language support is managed within a content item; while the device and re-branding support is managed by associating different Layouts (page definition layers) with a content type (item Template).

### *Multi-language Support*

Defining support for a new language is a simple process:

- Add a new language item to Sitecore Language list. Complete the ISO codes and dictionary reference.
- Space for language specific content is now available for each content item within the repository. Editors, given that they have permission to the language, can select the language they want to edit for a specific content item.

Each content item in the repository logically contains each defined language. When Sitecore renders a site it uses the default language (i.e. "en-US") that is associated with the website. By invoking a .NET or an XSL call, the default language is quickly changed, thereby rendering the any page in the selected language. There is no need to duplicate the site hierarchy, pages or content with this mechanism.

In addition, each content item language instance, though logically associated with a single content item, maintains its own version track. For instance, this allows independent versioning of language content per content item, such as German content at version 3, French at version 5 and American English at version 2. When new content is created for a language of an item, workflow is started for that language content of that item.

### *Device and Site Re-branding Support*

Sitecore can associate and automatically invoke various web page presentation formats for the same content item. Sitecore uses a concept of a logical Device that can be invoked automatically via browser agent and domain, or explicitly via a query string. Devices allow the same content to be presented in multiple formats, such as for web browsers, RSS, PDAs, alternate brands or search engines.

Devices are associated with content item types (item Templates). When a content item of a specific type is requested, Sitecore identifies what Device should be invoked for that content, which will automatically renders the content item with the associated Layout.

Again, this approach eliminates the need to duplicate the site hierarchy, pages or content, while providing complete separation of content with it associate presentation logic.

## Quick Analysis

Sitecore provides a true separation of content and presentation logic, thereby providing the ability to render multi-language content in a number of different formats, styles and looks without any duplication of content or pages. This enables designers and developers to implement site-wide changes to presentation logic, security, workflow and business policies quickly, as these attributes are fully independent of the content.

In contrast, SharePoint copies the site hierarchy, page definitions and content to a site Variation. This independently managed, with the possibility of evolving independently of the source site in terms of structure, content and presentation. This will create an undue management burden on editors, webmasters and administrators, as they are now must maintain physically separate sites, rather than generating new variations dynamically as done by Sitecore.

### 2.5.2 Navigation

**SharePoint 2007**

SharePoint uses a pluggable navigation model based ASP.NET Site Map Provider and Site Navigation mechanism. The default navigation system in SharePoint is based on the site hierarchy. The system generates dynamic page navigation based on the structure of the portal, with the site hierarchy defining the levels in the navigation menu. ASP.NET development is required to create custom navigation.

**Sitecore CMS**

Sitecore navigation, like DHTML menus, list menus, site maps and breadcrumb trails, are based on the information architecture of the website, which is the structure and relationship of the XML content items in the repository.

Content-level security automatically controls what parts of the content tree are trimmed and therefore what navigation is rendered. Navigation is quickly and commonly implemented as XSLTs or .NET controls that traverse the content items in the repository to dynamically generate navigation elements.

**Quick Analysis**

Sitecore enables web developers to implement a dynamic navigation quickly using standard XSL or .NET code. Many navigational elements, like a breadcrumb trail, take little as ten lines of XSL to define as reusable and skin-able element.

Creating a new navigational element within SharePoint requires a .NET developer to become familiar with the ASP.NET TreeView and Menu navigation controls that may be modified and extended to support specific functional and presentation requirements.

In addition, SharePoint developers will need to understand the ASP.NET Site Map provider model to develop custom providers that support specific business policy and security requirements. This will create a significant learning curve and require .NET development resources to implement a custom, yet common element of any website.

### 2.5.3 Content Reuse

**SharePoint 2007**

SharePoint supports a feature named "Reusable Content" that enables a user to maintain reusable content fragments in a centralized list for each site collection. Page authors can then reuse these content fragments in different Web pages.

A page author may add this reusable content to a page within the HTML editor. If subsequent changes are made to the source of the reusable content in the centralized list, the pages containing the references will be rendered with the latest updated version.

**Sitecore CMS**

Sitecore provides content reuse both within sites and across sites. Content reuse can occur at the presentation layer through content Renderings that assemble content on-demand. Content Renderings can pull content from any content items in the repository, even mixing-and-matching fields from different item types within the same Rendering. By default, content Renderings pull the current version of content thereby guaranteeing the reused content is always current.

In addition, content items may contain fields that reference one or more content items, again of any content type. This enables content editors to reuse content, like existing links, documents, FAQs or addresses, by associating them with a content item. A Rendering will render both the immediate and associated content items on a web page.

## Quick Analysis

Sitecore enable content to be reused in a mix-and-match manner from anywhere within the repository. Reused content is referenced which guarantees the content's current version and default language are rendered automatically. Sitecore provides a simple, consistent model for content reuse that enables unlimited flexibility.

SharePoint provides special mechanism for content reuse that forces users to maintain a list of reusable content. This mechanism, though providing rendering of the current version, offers no opportunity to reuse content outside of the current Site, let along across Site Collections.

# 3 Appendix

### 3.1.1 New and Improved SharePoint Features

| SharePoint 2007 Features | Business Value |
|---|---|
| Portal | Features useful for designing, deploying, and managing enterprise intranet portals, corporate Internet presence Web sites, and divisional portal sites. |
| Search | Consistent and familiar search experience, increased relevance of search results, new functions to search for people and expertise, ability to index and search data in LOB applications, and improved manageability and extensibility. |
| Content Management, including Document, Records, and Web Content | Delivers enhanced authoring, business document processing, Web content management and publishing, records management, policy management, and support for multilingual publishing. |
| Business Processes | Helps organizations streamline forms-driven business processes with easy-to-use, intelligent, XML-based electronic forms that integrate smoothly with existing systems. This security-enhanced, client/server platform provides rapid-solution creation and deployment, centralizes form management and maintenance, and helps to extend business processes to customers, partners, and suppliers. |
| Business Intelligence | Provide Web and programmatic access to published Office Excel spreadsheets, programmatic reuse of critical LOB data, and easy development of Web-based BI dashboards that can incorporate rich, data-bound KPIs, Web Parts, and published spreadsheets. |

### 3.1.2 SharePoint 2007 Improvements Inherited from Windows SharePoint Services v3 (WSS v3)

| Improvements to WSS v3 | Business Value |
|---|---|
| ASP.NET 2.0 | Leverage the ASP.NET natively from within SharePoint—remove COM/ASP dependencies. |
| Windows Workflow Foundation (WFF) | Provide a robust workflow engine used by content and business processes within SharePoint. |
| Pluggable authentication | Enable forms-based authentication (FBA) to leverage any directory of credentials. |

| | |
|---|---|
| Large-list support | Enable SharePoint lists to exceed 2000 items and comfortably support 100,000+ items. |
| Office 2007 integration | Store Office 2007 documents and their metadata natively within SharePoint. |

### 3.1.3 SharePoint 2007 Technical Resources and Descriptions

### 3.1.4 PowerPoint Documents

SharePoint 2007 Introduction:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/A201%20-%20SharePoint%202007%20introduction.ppt

SharePoint 2007 Multi-language support:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/A203%20-%20SharePoint%20sites%20multilingues.ppt

SharePoint 2007 Workflows:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/**A204**%20-%20**SharePoint**%20Workflow.ppt

SharePoint 2007 Web Content Management:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/A206%20-%20SPS%20Content%20&%20Web.ppt

SharePoint 2007 WSS v3:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/A208%20-%20Etendre%20SharePoint%20avec%20ASP.NET.ppt

SharePoint 2007 Introduction:
http://www.00001001.ch/Download/TechDays_Geneva/Applications/A201%20-%20SharePoint%202007%20introduction.ppt

SharePoint 2007 WSS Development and Custom Site:
http://www.toddklindt.com/blog/TechEd%202006%20Slides/OFC308__20060606_195517.ppt

SharePoint 2007 Dot-com meets SharePoint: http://blogs.msdn.com/ecm/attachment/631362.ashx

### 3.1.5 Blogs

SharePoint Team Blog: http://blogs.msdn.com/SharePoint/default.aspx

The Boiler Room Blog, Mark Kruger, SharePoint MVP:
http://www.SharePointblogs.com/mkruger/archive/2006/05/25/7570.aspx

Tom Stegeman SharePoint 2007 Blog:
http://www.SharePointblogs.com/tonstegeman/archive/2006/05/25/7583.aspx

Andrew May's SharePoint 2007 Blog: http://blogs.msdn.com/andrew_may/default.aspx

### 3.1.6 Diagrams

Using Columns and Content Types:
http://www.officezealot.com/downloads/moss/ContentTypesandColumnsBeta2Conceptual.pdf

Using Content Types in WSS v3:
http://www.officezealot.com/downloads/moss/ContentTypesBeta2Conceptual.pdf

### 3.1.7 Other Technical Documentation

Managing Web Content in SharePoint 2007: http://msdn2.microsoft.com/en-us/library/ms573556.aspx

SharePoint Server 2007 for MCMS 2002 Developers: http://msdn2.microsoft.com/en-us/library/ms406043.aspx

WSS v3 Glossary: http://www.microsoft.com/resources/documentation/wss/2/all/adminguide/en-us/stsglos.mspx?mfr=true