MODX Revolution Building the Web Your Way

A Journey Through A Content Management Framework

W. Shawn Wilkerson

MODX Revolution Building the Web Your Way: A Journey Through a Content Management Framework

Published by Sanity Press, a Sanity LLC company (sanitypress.com), Daytona Beach, FL 32117.

Library of Congress Control Number: 2012914262

ISBN-13: 978-0985853204 (Sanity Press)

ISBN-10: 0985853204

Printed in the United States of America

First Printing: August 2012

07 06 05 04 4 3 2 1

Bulk Sales: Sanity Press offers discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact

Sanity Press

1-386-322-7999

sales@sanitypress.com

Abbreviated Contents

Contents

Part I: Foundations

Part IV: Development

About the Author

W. Shawn Wilkerson has been around most every aspect of the computer and web industries for almost three decades. His client list includes on-line gaming clans, small businesses, ministries and non profits of various sizes, international corporations, and most everything in between.

After two decades in the industry, he returned to school and achieved nine Associate of Science degrees in computer and internet technologies in five years. On the advice of two department chairs and a multitude of professors, he decided to pursue business degrees attached to technology, as the common opinion was the course content in the Computer, Networking, and Internet fields were beneath his skill set and would continue to be so, for quite some time.

In May of 2011, he was awarded a Bachelors in Applied Science in Supervision and Management, which prepares graduates to work for any international corporation on the planet. The BAS also gave insight into the clients perspective, including the uncertainty and fears many of them share when considering the unknown technology universe.

During the Summer of 2012, Shawn began his Masters in Technology Management and expects to graduate in 2014, probably with honors as he has for his ten previous scholastic endeavors.

Over the years, he has programmed in a multitude of languages, beginning in the days of COBOL, RPG, and FORTRAN. Eventually he moved on to C, C++, C##, .net and many others. He finally made the jump to web programming and on a typical day of development, is known to combine as many as six languages into a single application - many of which utilizing reusable code since his move to MODX Revolution in 2004.

He began writing articles for his personal web site (`shawnwilkerson.com`) and was asked to add it to the official MODX Documentation and eventually joined the official MODX Documentation Team. At the time, he was able to spend quite a few hours in IRC and in the forums helping others work out issues with early versions of MODX Revolution, which didn't go unnoticed.

As a result of his contributions and content on his site and in the official Documentation, a major book publisher contacted him with a concept they were working on for a new line of books. Eventually, this did not work out as the company in two other markets for the publisher, causing them to drop the rest of the projects they had begun under their auspicious - including the three hundred page forerunner of this book.

After spending three decades working with clients and almost two decades working with on-line technologies he has performed every job related to web production from server administration to project management and typically lives in the midst of a few thousand lines of code.

His unique style of writing combines mental imagery and plain common sense.

Dedicated to

Jesus Christ, without whom I would not be alive nor have any purpose.

Victoria Wilkerson, my wife, partner and best friend. No one else could possibly fill your shoes or take the place in my life you have. We have come so very far together and it keeps getting better. There is truly gold and other very rare minerals to be found in your heart and your life. Vickie, my princess, I love you so very much - savf.

My kids: Christina, Joshua, Kymberli, Heather, and Chris. My I continue to discover what you need and better serve you as a father, friend, and mentor - for as long as you will have me.

Monica Wilkerson, my mother, who allowed me to purchase my first computer: TI-99/4A, get on line with CompuServe in the 1980s, and who coaxed me into AFJROTC where I would meet the woman who be more to me than simply a wife.

Michael Wilkerson, my brother, for ever holding true to the ideal of family. Maybe now he'll understand how there is "so much to say about a computer program."

Jim and Becky Pelletier, life-time friends and Pastors, who continuously pay a very high cost so others can find their place in a very dark and cruel world. Thank you for taking the hit for wounds caused by others in the "ministry" and restoring as many who will come.

Acknowledgments

For building an amazing platform and allowing the rest of us to use it, I would like to formally thank the following individuals who spend countless hours building, expanding, and extending their platform as far as the technology will go:

Ryan Thrash, MODX, LLC CEO

Shaun McCormick, MODX Senior Developer

Jason Coward, MODX Chief Architect

Jay Gilmore, MODX Director of Marketing and Communications

The MODX Community

Preface

As having had decades of opportunity to fulfill many of the roles related to the computer and web industries, I hope to provide a resource which directly benefits site owners, designers, developers, novices and professionals alike – albeit some as a training manual and others as a quick reference. Keeping in step with MODX Revolution thinking, readers of this work can travel as far as they are willing - individually or as a team. The work should prove beneficial to a very wide audience with varying levels of experience, expertise and needs:

- Individuals who understand the basics of HTML but have little or no prior experience with Content Management Systems, Frameworks, and/or Platforms
- Those who wish to have a well rounded tool-set, which can grow with them
- Individuals who build extensive web applications, but would like to be able to build "into" an existing platform and reduce their overall development cycle
- System Analysts and Team Leaders looking to build a cohesive Web Team

University professors should be able to use this text as a foundation in which to work with students in their capstone, project management, and other web related classes. The structure of the first section is ideal for this purpose and can be conducted by establishing a goal, assigning each of the corresponding chapters to a team member, and selecting a team leader to "manage" the project taking the position to build an entire project as described in the *Quick Start* chapter. More advanced students could oversee many of these "managers" and their teams.

Developers and designers may even decide to provide a copy of this book to each of the members within their various teams as a basis of interaction. *MODX Revolution: Building The Web Your Way* provides a foundation for each person to better understand their own roles, as well as enabling each of them to perceive that role and its significance within the whole project.

MODX Revolution, by its very design, can be implemented to bridge many of the gaps found throughout the web industry. In many situations, these gaps are simply issues of tech-culture and an inability to communicate. While discussing various aspects of MODX Revolution, suggestions, perspectives, and experience may be found, which could serve in making each member more effective as they fulfil their role in the project.

I personally believe, overall efficiency within a web project team can be directly traced to their ability to inter-operate. Not only should each team member be able to expand their own skill-set, they should also be able to effectively interact with the various other members and components of a given project. The ability to coordinate efforts to where one individual or team can build directly towards the needs of the receiving individual or team can prove invaluable.

Purpose for this book

Essentially, there are very few who will read this work, who I have not had the pleasure of having had spent some time in shoes very similar to theirs. Hence, the main purpose of this work is to allow people to take a few steps in my shoes, sharing some of the experience and understanding I have acquired over the years with MODX Revolution.

This book is intended to introduce topics to the reader and also walk with them just far enough to where they can develop and try new things on their own. By no means should this work be considered the final authority, nor can any book be - especially with MODX Revolution which is extensible in every way. I have spent seven years working with this amazing product and simply desire to help others do the same.

My ultimate goal is to facilitate communication and organization while maintaining flexibility and security -- essentially help others be more productive. I am not here to preach or judge anyone, though I do have my own opinions which occasionally are on the strong side, but usually for good reason. Hopefully, I will meekly and gracefully share insight which is considered practical and valuable.

Where I fail to do this, please forgive me.

Audience and Scope

As with all technical works, we could diverge our attention in many different directions, eventually diluting the material into oblivion. I have had to make some assumptions as to the audience and have had to limit the amount of peripheral conversation to those topics which would directly facilitate the utilization and implementation of MODX Revolution.

As to audience, it is assumed that the reader has *some* experience with web site design and is somewhat familiar with (X)HTML and CSS. Previous experience with Content Management Systems is not required, although previous experience with MODX products may provide a reduced learning curve.

The ability to code in PHP is not necessary to implement MODX Revolution for many web project scenarios, but this knowledge or having access to someone who is willing to learn the API and produce very streamline code will strongly assist in the creation of dynamic applications. In those areas where we move into examples utilizing PHP, this assumption extends to having *some* experience with PHP also, specifically versions 5.2 and above. It is my intention to provide content that is easily understood by a wide range of people, even in **Part IV: Development**.

As this work centers around the implementation and interaction of various web technologies, actual instruction concerning (X)HTML, CSS, and PHP will be left to other venues. Having said this, it should be noted the developers of MODX Revolution went to great lengths in their design

to ensure usability at just about all experience levels. *MODX Revolution: Building The Web Your Way* shares this goal in allowing the reader to get the information they need, without forcing them into areas they are uncomfortable with -- though we will get into more advanced topics as we go along as discussion in the latter parts will focus on implementation and less on explanation.

Each chapter begins with foundational concepts and may very well grow increasingly more complex as the examples and content is presented. Topics previously discussed and referenced in later chapters, will only be mentioned with regard to the relevant information necessary for the current example leaving the reader to return to the earlier content for more information. Feel free to use the index and table of contents to locate the topic where it is discussed in greater detail.

For those who would like to jump right in and start off running, feel free to jump to **Part II: Live Projects,** taking special note of the *Quick Start: Putting the Pieces Together* chapter, which moves very quickly through the various aspects of MODX Revolution, essentially demonstrating the various steps to quickly get a MODX Revolution project up and running.

Part 1: Foundations contains quite a bit of explanation and numerous examples regarding each of the topics necessary to become familiar with MODX Revolution. Each chapter presents a single subject, intentionally building in complexity, while incorporating concepts presented in earlier chapters. This section should be considered a reference on utilizing the Manager and a primer on the implementation of the various Elements utilized in web projects.

Part II: Live Projects moves into real world development examples incorporating jQuery, Ajax, Search Engine optimization, and various other topics for building better projects. In this area, I have endeavored to share concepts garnered from hundreds of projects. The topics presented may occasionally slip into the gray area of personal opinion and experience. Please feel free to treat these topics as introductory and not view them as the “only” methods to accomplish these tasks - especially with SEO.

Part III: Administration introduces the concepts and techniques to effectively administrate a MODX Revolution project. Access Control Lists and other “difficult” topics are presented in a straightforward manner utilizing very familiar terminology to quickly establish concepts usually considered to be advanced simply due to their assumed complexity.

Part IV: Development takes a “stone tossed across water” view of building dynamic PHP applications using MODX Revolution as a foundation. The areas of discovery range from the most commonly used methods in the API, to building third-party applications from scratch - including the xPDO schema and extending the generated classes into full applications.

Regardless of your skill level, this book was conceived with the idea of allowing you to begin at your current level of experience and continue as far as you would want to. The examples chosen for this book are intended to relate easily to a large number of readers. I intentionally refrained from using overly technical examples and verbiage so as to relate to a larger audience.

A word concerning Content Management

When choosing a Content Management application, project leaders should consider a few things: the perspective of the developers, the environment within the community, and how well the software matches the mission and vision of the company. Fortunately for us, the MODX developers purposely build applications which do not limit or overly define our interaction, and the MODX community appears to be set on pushing every boundary they can conceive.

Many Content Management Systems, seem to imply anyone is capable of building a web project. While this is true -- to a point, there is always a cost to master a skill and direct benefits which come *only* from doing so. What many web professionals take for granted, may have had an investment of years of college, combined with frustrated hours, days, or even weeks attempting to learn a new concept. This is magnified by MODX Revolution's ability to streamline work flows and package third-party content, which may inadvertently send the message what we do "is easy".

Throughout this work, many streamlined code examples and implementations will be discussed. If web professionals actually value their clients, they will not inadvertently communicate the simplicity of their involvement, skills, or tools. We should also not be unapproachable or exclusive. Project leaders should endeavor to make the client a part of the project, which is almost always mutually beneficial.

All too often, clients will take over their project and attempt to continue their development in-house -- usually in an effort to save cost. "After all, the web people make it look so easy." I have seen multinational corporations, clubs, and non-profits turn their sites over to a volunteer or their IT department only to watch the web site eventually get banned and removed from most of the search engines. The hard lesson for many of them: anyone can throw together a web project - only a few understand the "laws" of the World Wide Web which can make the site successful.

MODX Revolution can serve as the foundation for a company to become very successful and profitable, while also providing a platform which allows its users to start where they are and grow into new things. Additionally, it facilitates partnership with clients, by possibly involving them in varying levels of the process, helping clients to clearly understand the value of having experts on board to protect and implement their best interests and represent their business to the masses.

I do not pretend or assume to present the final answer in any of the topics discussed in this book. I am simply presenting techniques and concepts which have proven reliable over time. As you progress through these pages you will undoubtedly find multiple methods presented to accomplish similar tasks. It is my hope, you will be encouraged in discovering even more of them for yourself. Once you have acquired a few of these techniques, your production should greatly increase and continue to do so as more techniques are added to your skill-set.

Welcome to MODX Revolution, settle in and get real comfortable. This is a wonderful place to discover the very edges of the web, your skills, and come together as a very effective team.

3
Creating Project Templates

Often entire areas of cities are modeled around a general theme or ethnic group. In some of these, housing sub-divisions are erected using modular homes or cookie cutter homes. The buildings appear to be similar, but each home is actually defined by its occupants and has its own unique characteristics. Essentially, this is the purpose of Templates: to provide a consistent presentation, while allowing content to have freedom of expression.

Concerning Templates

Many modern web sites have been moved to the XHTML 1.x transitional or strict types, in hopes of being easily upgraded as new technologies are introduced. Currently, standards are rapidly moving towards HTML5 and XHTML5, so the temptation to begin implementing areas of the site using these technologies may be expected by clients. For now we will be using XHTML 1 and eventually HTML5 in later chapters. Designers essentially establish a basic foundation and set of boundaries for the site when choosing which Document Type Definition (DTD) to use.

MODX Revolution is standards neutral and as such, is not concerned as to the method utilized by designers to create web content – it simply puts the pieces together. Additionally, MODX Revolution does not create or produce any HTML or any other web language, in and of itself, it is only parsing content referenced through the various Elements and Components added to the respective web site project via the Manager. This affords designers the freedom to utilize any structured markup language currently available, as well as those still in conceptual states - even targeting different technologies towards specific areas of the project Who knows, someone out there may still require HTML3, which can easily be accomplished with a dedicated Template.

For the cutting edge designers, many modern document type definitions strongly encourage the separation of content from presentation. MODX Revolution Templates enable us to take this one step further, by facilitating the separation of page structure, content, *and* associated styling.

MODX Revolution Templates

MODX Revolution allows any text to be used as a Template and does not require that text to actually mean anything. A web page of complete gibberish can be created by designers, parsed, and displayed for the world to see. There are of course two "very minor" issues which may be of concern to some site administrators and their respective clients: 1) search engines typically ignore gibberish and 2) people usually do not enjoy watching their web browser crash. If having a web site actually visited by search engines or humans is of any significance to the success of the web site project, we would suggest using industry standard (X)HTML to design the page Templates.

Any generation of HTML and XHTML will work within MODX Revolution, though it is the author's professional opinion that most modern web sites should use XHTML-1.0-Transitional / Strict as the bare minimum. As we are discussing the utilization of a Content Management Framework, there are extremely few reasons to use page frames, and as such we typically avoid frames at all costs. It has also been our experience, Google and other search engines typically do not value frames very highly and in some instances, will ban a site using them from their search index. The examples within this book should adhere to XHTML-1.0-Strict standards or better.

Template Inheritance

MODX Revolution defaults to allowing any Template to be used in any location within a web project. That being said, it should be explained that when a new page is created it will inherit the same Template as the parent document for the new Resource. At the top (root) of the site this infers, the default Template, as defined in the System Settings (under the System menu), will be assigned to all pages created at that level, unless a new Template is manually assigned during the creation of the individual resource. Assuming a parent document is assigned a new Template such as "blueSkies", via the drop down menu available for that purpose when a resource is created or edited, then its children will also have "blueSkies" as their default Template.

This feature of MODX Revolution functions the way many web projects are built – everything within a directory utilizing the same presentation mechanism. Bearing this in mind, from the outset, could actually reduce the overall production time of a web site. It is by no means an enjoyable experience to manually edit dozens of child resources to change their respective Templates, even if someone were able to do an update query via MySQL based on the `parent`.

The main point here is to plan ahead. It may not be entirely unreasonable to duplicate Templates and use copies of the original in each directory or area of a web site. It is indeed possible, a few

months / years down the line, the project lead decides to have each directory themed differently, which could prove difficult to implement after the web site has "gone live." Flexibility often needs to be directly proportionate to the project's complexity and/or client size. Finding an efficient implementation of this flexibility can be a high-wire act – even on good days. Something as simple as the names utilized to refer to a specific Template or other Element can greatly hinder or facilitate a web project.

Naming Conventions

The acceptable *programming* practices associated with MODX Revolution requires the use of naming conventions which provide direct insight into function and purpose. Using this concept across all portions of our web project, even outside of actual programming, should prove highly effective to expedite its progress. By using colorful and descriptive names, non technical and technical users alike can more easily understand how things fit together within the web project.

MODX Revolution allows an optional description for many of the items created within the Manager, do so generously. Also, names and descriptions can be changed at any time.

The second major step for designers is typically to choose a name to refer to a Template. Previously we suggested a Template named "blueSkies". This may appear to be generic, but it could simply indicate the Template implements a blue theme based on sky imagery. Within the context of a specific web project it may be all that is needed. By understanding the purpose of "blueSkies", it should become apparent "greenForests" is a green theme centered around forest imagery.

For corporate web projects naming conventions should be considered even more critical. As sections of web projects are dedicated to divisions and their structures, it may become necessary to actually use the 50 character maximum length, as defined within the `MODX_site_Templates` table of the database. Often names such as hr_intake, hr_interview, hr_request, reps_order, reps_expenses, and reps_commissions are used to divide the Templates by department and function. This immediately aides in the site design, in that the valid use of a specific Template at a given location is simply indicated by its name. The Human Relations department can enjoy the simple prefix of hr_, whereas all content related to sales representatives can use reps_.

Ministries and non / not for profit organizations are usually more complex. Take membership for example. A church can have recognized members within their organization who regularly meet and interact. In terms of a web site, members can be from anywhere and not even directly related to the ministry itself. By implementing different Templates and naming them after the prospective groups, such as src_ for Sunrise Church and web_ for the members of a web site it should

help differentiate between the two groups. Imagine if these two member groups were mixed or reversed. Internal church data could be accessed, as well as possibly creating other privacy issues.

It should be noted, that MODX Revolution includes facilities to handle groups of users as well as documents which directly relate to many of these issues. By creating names which communicate purpose and function, it provides an additional tool to verify the correct implementation of a component within the web site.

MODX Revolution Rule #3: ***Choose names indicating function and purpose.***

Creating a Template

The process of creating a Template, for use within a MODX Revolution project, is typically straight-forward. For (X)HTML and CSS gurus this process can take just a few short minutes. Others may take much more time to get their first Template together. Many will revise and adjust a Template until they have it functioning as they have envisioned. Over time, an astute designer will have a hand full of Templates available as a starting point for most sites. Additionally, with experience and a broader understanding of MODX Revolution, these Templates should become more streamlined and standardized in their actual implementation.

Any typical text editor can be used to create Templates – free or commercial. Some will even choose to utilize the Templates included within their hosting package. Others may search the web for hours to find that perfect Template. For users of Adobe Dreamweaver, the process to convert the preexisting Templates to fully-functional MODX Revolution versions is extremely easy. A simple understanding of what is required to get a page working is all that is necessary.

As with many Content Management Systems, MODX Revolution utilizes standard (X)HTML Templates with formatted tags placed in key positions which are dynamically replaced with content and thereby result in a page presentation. Using these tags, designers are provided with immediate access to MODX Revolution configuration settings, resource settings, and user information for utilization within a web page. MODX Revolution provides hundreds of these keywords with the vast majority being optional. To best illustrate Templates, it may be best to look at a working example.

The default install of MODX Revolution provides a minimal Template which will serve nicely to introduce us to two of the types of document tags utilized within Templates and HTML `[[$chunks]]`. Hopefully, this section of anti-dramatic web page code is very familiar with the expected exclusion of the tags specific to MODX Revolution. As with many Content Manage Systems, much of the content is defined elsewhere and combined into a single tag.

Listing 03.01: default MODX Revolution Template

```
<html>
   <head>
         <title>[[++site_name]] - [[*pagetitle]]</title>
         <base href="[[++site_url]]" />
   </head>
   <body>
         [[*content]]
   </body>
</html>
```

Resource Tags

Recall that a document in MODX Revolution is called a Resource. This naming convention was selected based on the internationally accepted and standardized Uniform Resource Identifier (URI) nomenclature which specifies: "*a string of characters can be used to identify a resource via its location (URL) and its name (URN)*". This should help clarify why documents, Static Resources and Weblinks are all referred to as Resources, as each of these can be accessed via the address bar of a web browser.

The default Template code introduces us to the resource tags: `[[*pagetitle]]` and `[[*content]]`. These particular tags are always associated with the current document being displayed in the web browser and originate as a new Resource is created from within the Manager. These values are stored in the `MODX_site_content` table allowing many of the column (field) names in this table to be directly accessed via the *field_name formatting demonstrated above. For simplicity, each respective field is consistently named within the Manager, the database, and as Resource tags.

It should be noted, that many of the fields are optional and may return empty if the field is, in fact, empty or comprised of a NULL value. Other fields are reportedly used for internal MODX Revolution functions and may not have a corresponding Resource tag, but are still available programmatically. For the time being, having access to the `[[*pagetitle]]` and `[[*content]]` will provide the majority of page content within most sites. The minimum Resource tag required to make a web page functional within MODX Revolution is the `[[*content]]` tag. All other resource tags are optional, though some are highly beneficial.

System Settings Tags

MODX Revolution is capable of running multiple domains, sites, and other derivatives within a single instance. It should be noted that the primary difference between System Tags and Resource Tags is an issue of scope.

System Tags are uniformly the same across an entire Context (defined in Appendix A), with each Context capable of having it's own set of System Tags as it overrides the base System Settings. Resource Tags are always related to that Resource – regardless of which Context it is found in.

In the last chapter, we were introduced to the concept of System Settings noting that the default installation of MODX Revolution has approximately 150 System Settings. We demonstrated a method to change the name of a web project and indicated that the new name is stored as a variable in the `MODX_system_settings` table of the database. Many of these settings can be directly accessed following the naming convention demonstrated with `[[++site_name]]`, as all Systems Settings are prefixed with `++`.

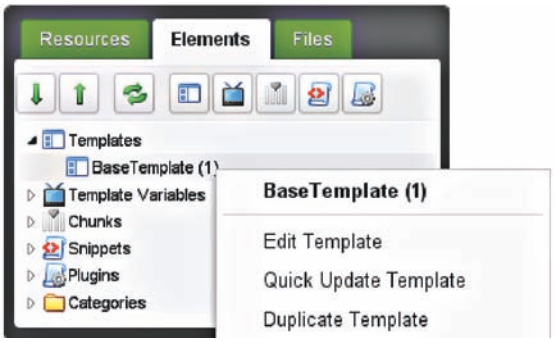
An additional difference between the System Setting Tag and the Resource Tag, is that none of the System Tags need to be included within a document for it to function. Both `[[++site_name]]` and `[[++site_url]]` could simply be replaced with straight text: Shawn's Website and `http://www.shawnwilkerson.com/` respectively.

(X)HTML Tags

The majority of the default Template is straight W3C - brand HTML, which is typical of many of the Templates implemented within MODX Revolution Sites. Any HTML or (X)HTML tag specified by the W3C is usable, *keeping in mind that not all web browsers are created equally*. Periodic testing of Templates with test content should be mandatory, as well as validation testing. Web sites such as `http://browsershots.org/` are great for letting designers see what a client sees. The W3C provides an on-line validation system, which can be found by visiting `http://validator.w3.org/`. There are also a host of other sites, in addition to similar functionality of many IDEs, which will validate and analyze the final page as well. It should be noted: the default Template should be edited, as it does not validate in its present form.

All Templates should implement the <base href> declaration, either with <base href="[[++site_url]]" /> , <base href="http://domain.com" />, or <base href="/" />.

Storing Templates in the Manager



The default mechanism for working with Templates is located within the MODX Revolution Manager. To access this feature we first need to log into the website by visiting yourDomain.com/manager. Once inside the Manager, access the Elements tab located in the center of the Site Component and Resource Trees (left side of the display area).

Next we open the Templates tree by clicking the arrow next to the Templates option. In a default install, this should scroll down and provide an interface similar to the one indicated. By right-clicking the "BaseTemplate" option, we are presented with a context menu providing each of the options available regarding Templates. Go ahead and left-click "Edit" in the context menu or "BaseTemplate" in the Element Tree which will take us immediately to an Editor where we can view the default Template. This can also be accomplished by simply left clicking the name itself.

Text Editors

By default, MODX Revolution includes a plain text editor, as shown when viewing the "BaseTemplate." Initially, I would recommend leaving the default Template untouched, until you master creating MODX Revolution Templates. Optionally, the CodeMirror Package can be installed via the Package Manager in the System Menu, which provides some basic functionality.

If you prefer an editor on your computer and find yourself limited to notepad or some other simplified editor, visit the MODX Development Environment information page, located at `http://rtfm.MODX.com/display/revolution21/Setting+up+a+Development+Environment` to find a list of editors tested and used by the members of the MODX community.

Syntax highlighting is the main feature of interest here, but additional features like code completion, code suggestion, verification etc., are also nice. The more features the better, especially if they can be turned or off, as it makes reading and editing code much easier during those 18+ hour days. Please keep in mind, it may be wise to create a local backup of the default Template, as well as any other Templates created for web projects.

For future reference, the Template name is used within the Manager for display purposes only. The identification number displayed next to the Template name in the Element tree, is the only reference stored in the `MODX_site_content` table in the database. This means a Template can be renamed as much as desired without interrupting the web site or inadvertently breaking pages due to a naming mismatch.

Example XHTML 1.0 Strict Template

To get started we will begin by creating a new, somewhat typical, MODX Revolution Template for use on our site. We will be needing access to the Manager, so feel free to keep a Manager tab open in your web browser. Select Elements to move into the Element tree, right-click Templates and select Create a New Template and enter the following, saving it when finished:

Listing 03.02: RevoBook Template

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
        <meta http-equiv="Content-Type" content="text/html;
                charset=utf-8" />
        <title>[[++site_name]]'s [[*longtitle]]</title>
        <base href="[[++site_url]]" />
        <link rel="canonical" href="[[~[[*id]]]]"/>
        <meta name="author" content="W. Shawn Wilkerson" />
        <meta name="copyright" content="Copyright ©2010 W. Shawn
                              Wilkerson, All Rights Reserved" />
        <meta name="date" content="[[*publishedon]]" />
        <meta name="description" content="[[*description]] by W.
                                          Shawn Wilkerson" />
        <meta name="keywords" content="word1, word2, word3" />
        <link type="text/css" rel="stylesheet" href="revo.css" />
  </head>
  <body>
   <div id="header">
   </div>
   <div id="container">
       <div id="mainContent">
           [[*content]]
       </div>
       <br class="clearfloat" />
       <div id="footer">
       </div>
   </div>
  </body>
</html>
```

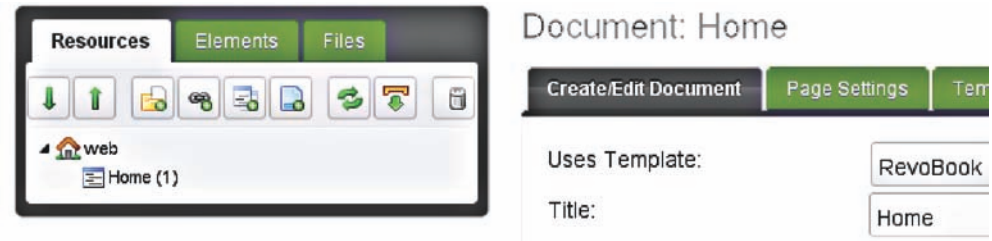
Listing 03.03: RevoBook HTML5 Template

```
<!DOCTYPE html>
<html>
<head>
    <base href="[[++site_url]]"/>
    <meta charset="utf-8"/>
    <title>[[*pagetitle]]</title>
    <meta name="author" content="W. Shawn Wilkerson" />
    <meta name="revised" content="W. Wilkerson, [[*publishedon]]"/>
    <link type="text/css" rel="stylesheet" href="revo5.css" />
    <link rel="canonical" href="[[~[[*id]]]]"/>
    <!--[if IE]>
    <script src="//html5shiv.googlecode.com/svn/trunk/html5.js">
    </script>
    <![endif]-->
</head>
<body>
<header>
    <nav>
        [[- global navigation - this is to be a MODX Revo Comment ]]
    </nav>
    <div id="header_middle">

    </div>
</header>
<section class="main_content">
    <div id="container_wide">
            [[*content]]
    </div>
</section>
<br class="clear">
<footer>
      [[ - global footer element will go here]]
</footer>
</body>
</html>
```

W3C Validation

Before we do anything with this Template, we should probably take a minute to validate it. Using the left mouse button, click on the Resources tab, and then on Home. We are now going to change the Template for the Home page to use RevoBook instead of BaseTemplate. Simply use the drop down bar to Select the new Template. If it does not appear, there is a very distinct possibility the Template was not saved and the creation process will need to be repeated. Once the new Template is in place, save the document and select View.



Before panic sets in, remember that we have not began creating web pages or content and the default "Home" is indeed empty, hence, the bland, white, empty page. Copy the URL from the web browser address bar and use the W3C validator by visiting `http://validator.w3.org/`. Note: HTML5 validators are still a work in process. The page should correctly validate to XHTML Strict and provide a real friendly green "Passed" rating. Without any errors or warnings in our Template, this is a good starting point to begin working on the rest of the site.

Anyone who wants to live on the edge can simply type their name or whatever comes to mind in the content area of the Home Resource, save the document and then View the page. You should now have content appearing on the front of the web site. If you want to go all the way and leave nothing for the return trip, validate the web page with the new content and see if you continue to achieve the coveted green "Passed" recognition from the W3C. At this stage it may very well be prudent to print out the RevoBook Template and the source code from the preview of the "Home" page. Take interest in the changes between the various `[[*resource]]` and `[[++system]]` tags used in the RevoBook Template and the final page parsed by MODX Revolution. It also might be useful to use landscape page-layout and a color printer, if your editor supports this functionality.

In the web projects at my company, the use of any on-line based Rich Text Editors is strictly frowned upon – at least as far as designers and developers are concerned. Clients are another matter. There are three main reasons for this: 1) Any updates, overwrites, deletions, and other activities to sites are currently permanent. Once an Element is saved there is no going back. 2) If another Manager user (or ourselves) accesses and destroys our work of genius, we would have to resort to pulling it from a database backup – assuming there is one. 3) Rich Text Editors have a tendency to "fix" correct code. Note: versioning is in future plans for MODX Revolution.

To solve many of these issues before they turn into a perfect storm, we have established a company policy which requires all additions to the web site to be stored on a local system and backed up to a local server, before being put into MODX Revolution. There are many different methods used to have a local version of a web site. We would simply suggest: `type_name-date.ext` (template_Main-201104020.xhtml) or building sub-directories matching those used on the site or even using a directory to match the type of element being worked on (`templates\name-date.ext`). For us, each individual item is stored with a date after the name and before the extension and a combination of these solutions. Feel free to develop your own to suit your needs.

We also have another question which may eventually need to be answered, namely which Template is going to serve as the default Template of the site. Currently, the BaseTemplate is the default Template site wide. This can be remedied in a couple of ways:

1. Duplicate the default Template and rename the copy to something like MODX Default, followed by subsequently over writing the current BaseTemplate with our chosen default Template, and optionally renaming the Template as needed
2. Change the default Template setting in the System | Settings interface by using the drop-down menu.

Template Storage Alternatives

While slightly outside of the purview of this chapter, we should take a moment to mention a non-standard method of storing live Templates on a server.

Templates available via Package Management

We should probably mention, a number of Templates are conveniently available via Package Management located in the System menu. Click on Download Extras and select the Front-End Templates option near the bottom of the list. Be careful to honor the licenses attached to any MODX Revolution Addon.

Using Media Sources: Filesystem

Introduced in MODX Revolution 2.2 was the ability to store Elements using technologies and APIs from all over the web. Each medium can be defined in Media Sources and subsequently utilized through each Elements creation process. In regards to Templates, simply select the Is Static checkbox, and use the Static File (MODX Browser) to locate the file. This allows us to bypass using the Manager for subsequent modifications to the Template, as they can be simply uploaded via the IDE software, FTP, SFTP, SCP, etc.

You could also utilize your own cloud space to store and share Templates and other Elements.

Summary

MODX Revolution Templates can easily be imported via the Manager and provide a consistent presentation, while allowing content to have freedom of expression. Users may design web content using any available W3C standard, as well as any Document Type Definition (DTD) recognizable by web browsers. Multiple W3C standards can even be used within the same site. It is very important each Template utilizes the `<base href=" " />` tag.

Child documents will inherit the Template used by their parents, with the top level pages using the default Template for the site (or as defined via the context the Template is to be used in). By default, Templates can be utilized by any resource located anywhere within a web project and only need the `[[*content]]` tag to function. Many of the hundreds of MODX Revolution system and resource settings are available can greatly enhance and simplify the web design process, while predominantly remaining optional.

The names utilized to create Templates and the other components within a web project should speak to the function and purpose of the given item. Many of the MODX Revolution tags are so well named, that supporting documentation is seldom needed to ascertain their role within a web site. This habit should be developed and extensively exercised throughout the entire project.

Lastly, during this chapter we have been exposed to the vast majority of issues and topics related to Templates, while simultaneously weaving other topics into the discussion. In the process we managed to sneak something past many of you. Back in our RevoBook Templates, we stacked multiple Revolution tags together to create a variable which is changed on every page. This variable simply creates a link to the current page and is parsed from the inside out. First MODX Revolution gets the current document id, and then it creates a full path to it using the documents name: `[[~[[*id]]]]`.

This implementation represents a small sampling of the complexity which can be implemented with MODX Revolution using just its tags. Hopefully, this may be reason enough to explore these tags and to discover the power waiting to be found within MODX Revolution.

Index Of Code Listings

Note: All chapters are intentionally listed whether or not they contain code listings.

Index

Symbols

A

B

C

D

E

F

P

R

S

T

U

W

X