

# Best Practices for Implementing RedDot CMS

**nine lessons from the field**

Release Date: April, 2004  
Authors: Cathy Bath, Randy Woods



## Table of Contents

Preface .....	2
Introduction .....	3
But before we start.....	4
Section One: Avoiding Future Headaches - Decisions to simplify future maintenance and administration.....	7
Tip the First: Build a solid - and singular - foundation.....	7
Tip the Second: Organization is next to Godliness.....	8
Tip the Third: Making Content Classes and Templates Work.....	9
Section Two: Short Cuts that will Save You Time (But not cost you your job) .....	11
Tip the Fourth: Fight the right battles.....	11
Tip the Fifth: The name is the thing .....	12
Tip the Sixth: Stay plugged in .....	13
Section Three: Usability, Compliance and Reporting.....	14
Tip the Seventh: All Hail the Feedback Loop.....	14
Tip the Eighth: Seek Purity .....	15
Tip the Ninth: Perspective is Everything .....	16
About non-linear creations inc. ....	17

# Preface

nonlinear creations inc. is a leading integrator of content management solutions. The RedDot CMS solution is a key member of our partnership portfolio.

We have completed more than 20 RedDot CMS implementations – involving more than 250,000 pages of content – and we have learned a thing or two from this experience. This document pulls together the most important lessons we have gleaned from fighting real-world implantation battles.

We hope that you and your team will find this document useful. But we are even more hopeful that you will decide to give us a call and ask for our assistance. You can reach us by email at [randy@nonlinear.ca](mailto:randy@nonlinear.ca), or by phone at 613.241.2067 ext.234.

# Introduction

All content management systems hold out a set of gleaming promises – reduced IT involvement in content publishing, streamlined workflow and approval processes, simple rollback features and reduced costs of operation. In real life, getting a content management system to deliver on these promises requires you to make very good decisions, very early in the implementation process.

RedDot CMS is an ideal solution for the mid-market. It is easy to use and relatively straightforward to implement. But as is the case for all content management systems, you have decisions to make during implementation. Good decisions will make your life easy; bad decisions can wreak havoc for years to come.

Our RedDot implementation experience falls into two categories:

- Large scale implementations undertaken for international companies and government organizations
- RedDot “rescue missions” where we have been retained by clients to help them recover from a failed or failing implementation undertaken by other vendors or in house staff.

This has given us two types of insight. In implementing RedDot “from scratch” we have gained experience with the types of implementation decisions you will need to make. In rescuing failing implementations, we have witnessed the wreckage that bad decisions can cause. In this document, we attempt to summarize our experience in a set of straight forward recommendations.

We need to make one large caveat: this document focuses specifically on technical decisions made during implementation of a RedDot system. It is designed for technical and semi-technical staff faced with the challenge of making a content management system work.

For many organizations, success or failure is determined by questions of governance and content management process. We will be releasing publications addressing these issues in the coming months. But they are outside the scope of this discussion of technical best practices.

This document is divided into three sections:

## **Section One: Avoiding future headaches**

- These three tips won’t change your life today, but someone down the road will sing your praises. Or, at least, not take your name in vain.

## **Section Two: Work smarter, not harder**

- You can spend an infinite amount of time on any IT implementation – but none of us has that long. Three keys to using your time wisely.

## **Section Three: Making it work**

- It doesn’t matter how good the technology is, if it is never used – steps you can take to deliver real-world success.

## But before we start....

The remainder of this document assumes familiarity with the RedDot content management server. If you are comfortable with RedDot concepts and terminology feel free to skip this section. If, on the other hand, you have little or no experience with RedDot, than this brief overview of RedDot concepts should render the rest of this document at least vaguely intelligible.

### Variants

Like most mature content management systems, RedDot CMS allows you to separate content from presentation. In addition, RedDot allows you to separate both content and format from navigational structure. These variations are called, appropriately enough, “variants.”

- A “project” variant defines a site structure. The same content can be delivered to different projects. An example may clarify this. A company might operate both an extranet and a publicly accessible web site. In both cases, they want their press releases to be published online. By defining the extranet and web site as separate projects, RedDot can publish these news releases to two locations and into two completely different site navigation structures.
- A “template” variant defines the visual representation of the page into which content is inserted. The same content can be generated as an image-intense page suitable for viewing over high-bandwidth, a low-bandwidth variant designed for dial up access, and a wml variant suitable for viewing over a wireless device.
- “Language” variants allow you to maintain the exact same site in multiple languages. Each site will differ only in the language of content – not in navigation or site structure.

Importantly, templates can be “pre-assigned” for sections of the site. When an author creates a page in a section, they will be limited to choosing from the templates “pre-assigned” for that section.

### A “foundation” template.

RedDot documentation does not address the concept of a “foundation” template or page, but we find it to be a powerful and universally applicable concept.

A foundation template is the initial document from which you create all pages within a project. Each new page created from the foundation page template inherits the characteristics defined within this foundation template. As we note in the first tip below, you *can* create several foundation templates. But you probably shouldn't.

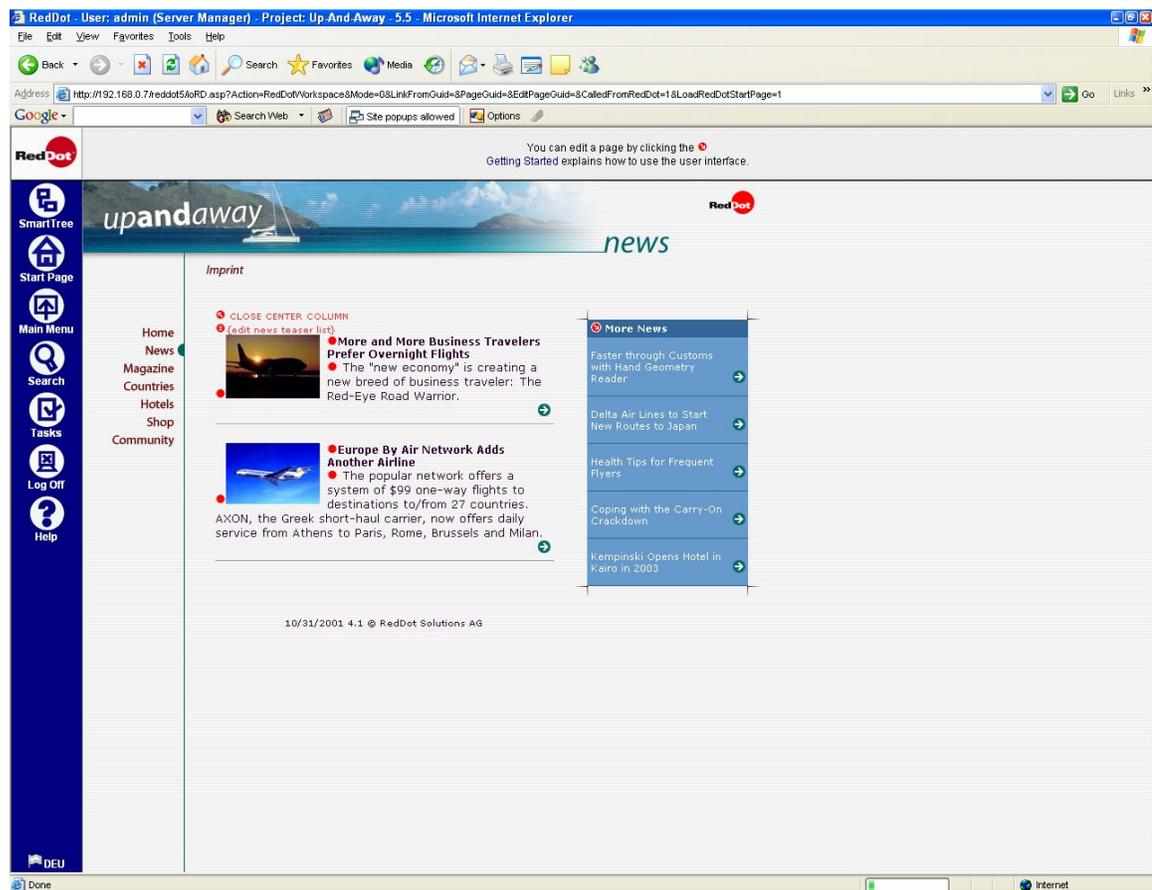
### Some RedDot Terminology

Like all software systems, RedDot has adopted specific terms to describe its features and interface. These are the most important

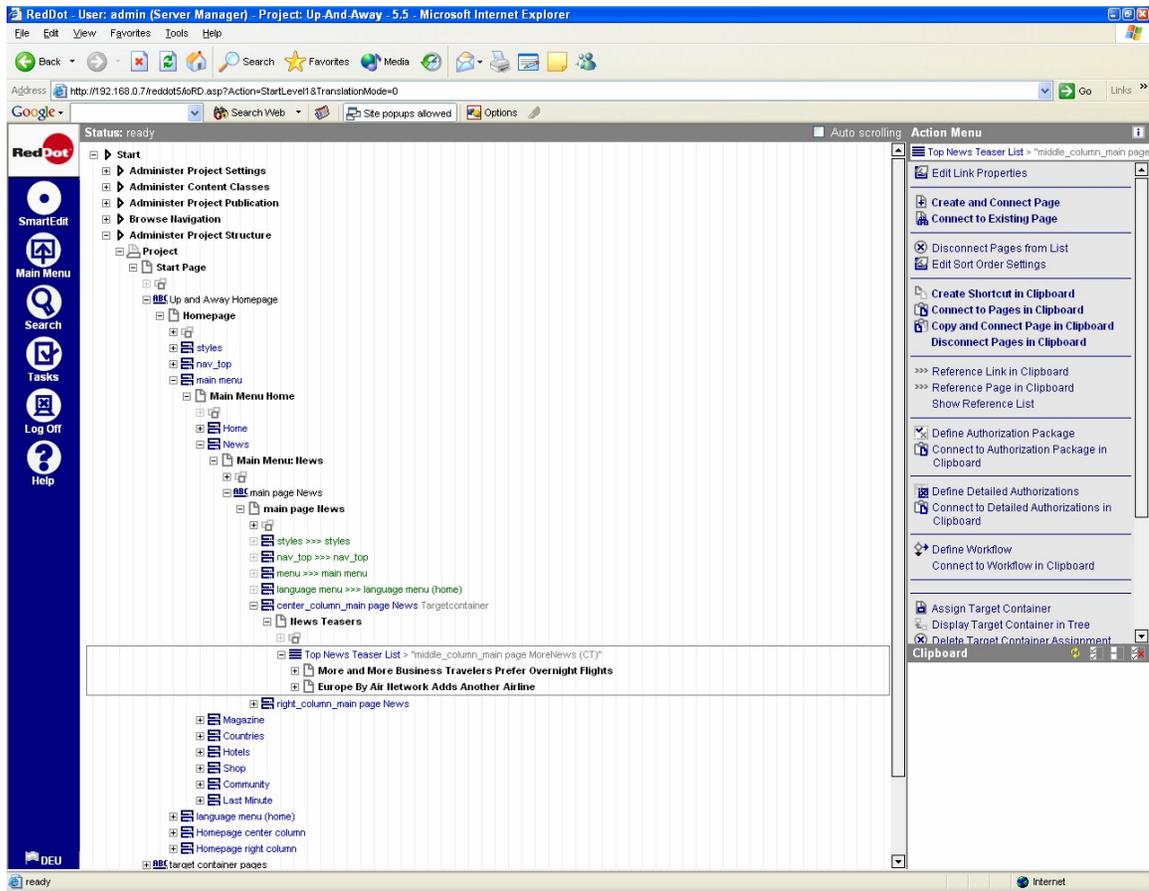
## SmartEdit and SmartTree

RedDot allows access to CMS-content via one of two interfaces – SmartEdit and SmartTree. SmartEdit was designed specifically to meet the needs of content creators and editors. SmartTree allows for easy, intuitive administration of the site.

The **SmartEdit** view allows authors and editors to browse the web site until they find the content they want to edit. As they browse the site, they will see “red dots” on any item that they have permission to modify.



In contrast, the SmartTree mode represents the site according to its logical structure. This “tree” navigation allows administrators to create and group templates or content, and to apply a variety of rules for any section or template.



## RedDot Query Language (RQL)

One of the great advantages offered by RedDot is the extensibility enabled by its XML-specified API. RQL allows a trained developer to integrate individual, external applications into the RedDot CMS, or to customize features or functions within RedDot.

By creating “plugins” using RQL and a server side scripting language such ASP, a trained developer can manage unique feature needs or integration requirements quickly and efficiently.

# Section One: Avoiding Future Headaches

## Decisions to simplify future maintenance and administration

### Tip the First: Build a solid - and singular - foundation

Not surprisingly, a solid foundation is critical to a successful RedDot implementation. The RedDot CMS allows you to specify a “foundation” template from which all pages will be derived. This template can be used to define content or content types that will appear on every page generated - a link to privacy policy, corporate logo or contact information are classic examples. An ideal foundation page is as simple as possible, allowing for maximum flexibility of the derived content templates.

#### The Challenge

RedDot developers frequently face a challenge – individual *subsections* of the site may require specific “mandatory” or “standard” elements. Elements placed in the foundation page become standard *across the entire site*. For example, a link to the privacy policy may be required on every page on the site, but contact information for the employee recruitment group may be a mandatory element of the web site’s human resources section

The temptation is to create a foundation page for each section of the site. This would appear to be a simple, and logical means for managing common, section-specific content.

But, as is usually the case, giving into temptation is a mistake. Potentially a big mistake. non-linear creations’ was recently engaged to clean up a large RedDot implementation. The original firm contracted to perform the implementation had given in to temptation – there were no fewer than twenty-seven separate foundation pages.

Our review revealed that the number of headaches reported by the staff responsible for administering RedDot was exponentially related to the number of foundation pages. These headaches had two immediate causes:

- Every time a change needed to be made to the entire site, the administration staff had to chase through each of the twenty-seven foundation pages. Each page needed to be changed and submitted to workflow for approval. Then released for publishing. And this had to happen in two languages in parallel. Using more than one foundation template eliminated most of the value of having a foundation in the first place. The result? Headaches for all concerned.

- End users were perplexed every time they created a new page. Rather than being presented with a nice, neat list of templates available for the section in which they were editing, they found themselves facing an ugly mixture of foundation page and templates. Failing a direct mind-meld with the original developers, these users had no way to choose a template save trial and error. And after a couple of trials and errors, they called the RedDot administrator. More headaches.

### **The solution**

So how do you address the need for section level common content? The answer depends on the number of languages in which the content is published.

For a unilingual site, you can create a section-specific template that includes the section-specific elements. Then - with the careful foresight provided in the next tip - you can restrict end users from making use of any other template within that section.

If the site is multilingual, the solution becomes more complicated. In most multilingual publishing situations, you will need to have matching content for each language. For example, all of the pages in a section may require a special title graphic. But the Spanish pages require this graphic in Spanish - the English pages, an English-language graphic. Simply using a section-specific template does not help - you need to create logic so that the system "knows" which graphic to display. Fortunately, RedDot supports this. In creating the language variant, you can define reusable, conditional code to deliver the right graphic in the right language

### **Summary: headache prevention strategies**

- Use only one foundation page.
- If the implementation involves content in only one language, use section-level templates to enforce section-level standards.
- If the site is multilingual, use RedDot's "reusable component" feature to drive section-level standards. Tie-in these reusable components by writing a plug-in using RedDot's RQL language.

## **Tip the Second: Organization is next to Godliness**

On more than one of our RedDot rescue missions, we have encountered implementations where all RedDot templates were "grouped" in one, large directory. We know how this happens - the original implementation team underestimated the number of templates they believed would be required. There seemed no reason to create an organizational hierarchy - administrators could simply read the template names and understand their purpose.

This approach works. For about two days. Then the demands of more content, and the discovery of the need for more templates, produce an increasingly impenetrable list. When the number of templates in the folder approaches twenty, they become essentially unusable - we've seen implementations with more than 200 templates in the same directory. Publication grinds to a halt and the demand for Tylenol in the RedDot administration group skyrockets.

**The solution**

Start with a simple organizational structure. Segment your templates into logical groups. We often start with a simple folder structure – content, foundation pages, navigation, scripts. Templates for specific subsections are grouped within new folders in the content section. Common navigational elements can be grouped in sub-folders of the navigation folder.

The resulting clarity is appreciated, not just by today’s administrators, but by those who inherit responsibility for the system in the future.

*How to:*

To add new template folders, select Administer Project Settings. Then select Folders and from the action menu select Create Folder. Type in the folder name, and select the type of folder as Content Class Folder. Select OK. Repeat for each new folder you will create

## Tip the Third: Making Content Classes and Templates Work

The concept of a template is straightforward: create a set of html that will form the structure into which non-technical authors can insert content. The advantages are many –templates make it easy to display the same content in multiple ways, enforce design and brand standards, consistency of navigation – the list is long. To realize the promise of templates, however, you need to make smart decisions during the initial CMS implementation.

**Pre-Assignment**

The standard implementation of RedDot CMS restricts how end users can create new pages. Each page must be created from a “link,” the link that will allow a visitor to access that page. For links within any area of the site, the administrator can pre-assign Content Classes. An example will clarify why this is an important concept. Consider an end user who is creating a new press release. They navigate to the press release section of the web site using SmartEdit and click on the appropriate “red dot”. To define a link, they select “create and connect page.” When they do so, they are presented with a list of content Classes. If the administrator has pre-assigned Content Classes to the “press release” section of the site, the end user will only see the Content Classes associated with the section – they will be forced to choose an appropriate Content Class.

If the administrator does not pre-assign Content Classes, the end user will be presented with a long list of all available Content Classes. Two problems result – the end user may be forced to hunt through many templates before identifying an appropriate structure; and control over presentation of content within that section is lost. Mangled content means unhappy authors and editors turn to the administrator for help.

Pre-assigning Content Classes during the initial CMS implementation avoids these headaches. This is a somewhat tedious task, but the investment of time pays big dividends over time.

#### *How to*

To pre-assign a template, expand the Structure Elements node under the Content Class. Select the appropriate link element (Anchor, List, Container). Then from your action menu, select Preassign Content Classes. A popup will appear that will list all Content Classes in your project. Select the check box beside each template that will be required for users to create/connect new content.

### **Adding Content Class Descriptions**

This would appear to be an obvious best practice. Our experience rescuing failed implementations suggest that it is not.

Without a concise description of the role of each Content Class, authors and editors must rely on the Content Class name for guidance when creating a new page. This is a bad idea – what seems a perfectly clear class name to one person may be incomprehensible to another. Clarity counts. Taking the time to add a clear description of each template will make everyone’s life easier after launch.

#### *How to*

You can add a Content Class description either when you are creating a new Content Class or after it has already been created. For a new Content Class, select the folder in which you would like your new Content Class to be placed, select Create Content Class With Template. Then fill in the Content Class name. Directly below the Content Class name is the description field where you can fill in the details of the class/template. If you neglected to add the description when you created a Content Class you can go rectify this at anytime by selecting the Content Class and from the action menu select Rename Content Class. The same name and description textboxes are available for updating.

### **Enforcing Good Taste**

If you have spent anytime working with a large number of authors, you will recognize that individuals taste in design, or lack there of, varies dramatically. One reason for using templates is to control your organizations’ image – preventing those with a penchant for fuchsia from “enlivening” your site, for example. But there is a “hole” in the template approach that allows the design-impaired to wreak havoc. By default, authors can format any text entered into RedDot text fields defined in the templates. This includes font size and, yes, colours. At NLC, we refer to this as the “fuchsia gap.”

Fortunately, there is a solution. The administrator has the power to “turn off” any or all formatting capabilities within a given text box in a given template. Based on our experience, we strongly recommend you control the font design and colour using cascading style sheets, and eliminate text formatting from most, if not all, templates.

## Section Two: Short Cuts that will Save You Time (but not cost you your job)

### Tip the Fourth: Fight the right battles

Before the advent of content management systems, “webmasters” took great pains to organize hand-coded html pages within a directory structure mapped to the meaning of the content it contains. For example, a corporate site might have an “about” directory, which contains a “media” directory, a “management” directory and a “corporate history” directory. The URL of a press release might read:

[http://www.corporatesite.com/about/press/2004/february/press\\_release\\_02\\_04\\_04.html](http://www.corporatesite.com/about/press/2004/february/press_release_02_04_04.html)

There is a temptation to ensure that the RedDot-powered site replicates this directory structure *exactly as it existed* before the introduction of RedDot. And you can make this happen – but it is a time-consuming process and often unnecessary. When content is organized within the RedDot CMS, the intuitive naming of a hierarchical directory structure for the live web server loses its importance. Organizing content *within* the RedDot system is a much better way to spend your time.

We have found that the simpler the publication structure the easier it is to maintain – and the faster it is to implement. In most cases, we create a very simple directory structure based on languages and content types. A sample directory structure might be:

- Root
  - English
  - French
  - Spanish
  - Images
  - Includes

### Language Variants

There is one critical exception to the rule of directory simplicity – each language variant of a page must be published to a different directory. RedDot creates *the same* page id for all language variations of the same content. This allows it to synchronize publication across languages and to monitor the workflow associated with translation. These are good things.

But, if the variants are published to the same directory, they will over-write each other – the last language variant published will be the only page available. This is clearly not good.

## Tip the Fifth: The name is the thing

RedDot gives you significant flexibility in naming the files the CMS system publishes.

Webmasters hand-coding HTML generally name files with reasonably easy to interpret names such as “index.html” “about\_us.html” “press\_release\_index.html”. RedDot allows you to continue this tradition. You can require each author to complete a field in the template specifying a file name for the page.

However, RedDot will also happily name pages automatically. By default, it assigns a file name based on a unique variable – the GUID. This is a lengthy number and it creates a URL only a programmer could love, like

<http://www.yoursite.com/content/63ECEB706F3B44CF8C98D9DEFC283F06.html>.

As an alternative, it can apply a consecutively numbered page id – [www.yoursite.com/content/7.htm](http://www.yoursite.com/content/7.htm) - a much cleaner alternative.

It is easy enough to force end users to define a field name for newly created pages. But defining field names for thousands or tens of thousand of pages when a WCMS system is first being implemented is no one’s idea of a good time.

If a page is heavily bookmarked, or widely-referenced by external sites, you want to avoid disappointing visitors. But you want to minimize manually naming pages that are not entry points.

Fortunately, a quick review of the existing site’s server logs will quickly tell you which entry pages are most important. Any number of log analysis tools can process server logs and identify key “entry” pages to the site. In virtually every case, a small number of pages will be the route by which the vast majority of visitors enter the site. Two examples from recent NLC implementations:

- On one large consumer-oriented site, 90 percent of visits originated at one of three pages
- On an important B2B web site, 50 percent of visits originated at one of fifteen pages (the next 85 most frequent entries points only increased this figure to 53 percent.)

### Best Practice for Manual Naming

At NLC, we follow these principles in naming pages within RedDot:

1. Analyze server logs and understand where visitors enter the site. Ensure that key entry pages are manually named. This may require the creation of directories specifically for these pages to ensure inbound links are not broken.
2. Allow RedDot to automatically name other pages on publication. But *avoid* the use of the GUID naming convention. External search engines, including Google, generally penalize long URLs, and the GUID can be very long. You will have better search engine results if you have RedDot name pages consecutively.
3. Enforce manual naming for new pages added to the site. It takes an author only a moment to name one page – and it will improve ease of URL interpretation going forward.

## Tip the Sixth: Stay plugged in

One of the key attractions of the RedDot CMS is its extensibility. “Plug-ins” can be created that extend the capabilities of the system to meet any unique challenges you might face. (One-caveat – “plug-ins” will only work with the “enterprise” version of RedDot, not the “express” version.)

RedDot provides a number of these “plug-ins” in the RedDot community site – [www.RedDotcommunity.com](http://www.RedDotcommunity.com). Access to this site is granted to individuals completing RedDot training. Go to the “software” section of the site and click on the “plug-ins” tab.

These are the “plug-ins” we have found most valuable:

Plug-in Name	Function
Mass Upload	The Mass Upload plug-in uploads an entire directory of files to the RedDot Server with a single click. Saves time when you need to move many files at once to the RedDot server.
Clear RedDot Temp Directory	The Clear RedDot Temp Directory plug-in deletes all files from the RedDot Temp directory on the RedDot Server. If you do not have file access to the RedDot server, this plug-in can greatly simplify your life. It lets you clear an entire directory than do a “clean” republic of the entire site. You do not need to worry about the creation of orphan files during the republication.
Reference Link in Clipboard to all pages of a template.	<p>This plug-in is particularly valuable when you realize you have forgotten to add a common component to all of the pages on a site (the privacy statement, for example.)</p> <p>Before inserting the new component, the plug-in will warn you that all previous references will be over-written. It also gives you the option to preassign the same reference in the template, ensuring all new pages will contain the component.</p>

### Create Your Own “Plug-ins”

If you know XML or ASP, the RedDot plug-in framework puts a wealth of flexibility in your hands. The Administration III course, available from RedDot, course focuses on RQL (RedDot Query Language) which is the API for all CMS functions within RedDot. With the knowledge you gain from this course, you can quickly customize implementations to solve challenges specific to your environment, content or user needs.

For example, NLC had a client who wanted to reduce the number of steps required to create a new page. RedDot has prioritized usability over speed – as a result, a end user may need to create a foundation page, then create a content area, then add a specific menu item on which a new page is created.

We created a one-button page creation solution using an RQL plug-in. The user clicks a button, a new page is created, and they complete a popup menu specifying page name and section. The “plug-in” managed everything else.

## Section Three: Usability, Compliance and Reporting

### Tip the Seventh: All Hail the Feedback Loop

There are a lot of functions to keep track of when you are implementing a large CMS project. If you miss one – or if you implement it in a way that your end users do not understand – you will pay for it in painful redevelopment and deployment.

That is why NLC pursues a simple, iterative approach during implementation. This approach has two stages:

#### Stage One: You are the Guinea Pig

You should be the first step in the feedback loop. Each mistake you catch is one fewer that end users will grumble about. We recommend that you:

1. Create one foundation page and a number of content classes (templates) specifically for the content area.
2. Switch to SmartEditor, the interface used by authors and editors, and try adding some content. You will immediately see red. Red Dots. But almost certainly not as many as you need. Try changing each “editable” element on the page – you will probably discover that you have forgotten something.
3. Make the appropriate changes and try again. Once you are happy with the results, go to stage two.

#### Stage Two: Release the Hoards

You need to find a group of friendly end users. Invite them to join you for a half-hour session. Pull up the handful of pages you’ve created in SmartEdit mode and invite them to edit the content.

Within 15 minutes or so you will have a short list of suggestions and a short list of requests for additional features. Taking the sensible suggestions and requests into account, change the interface appropriately.

This may seem like an unnecessary step in the implementation process. But the time consumed in this early informal testing pays huge dividends during the final role out.

## Tip the Eighth: Seek Purity

Many organizations adopt a content management system to enforce standards. These standards frequently involve compliance with defined HTML standards – a common example is the W3C accessibility guidelines.

RedDot allows you to enforce compliance with these guidelines when content is created within the RedDot interface. But end-users invariably “cut and paste” from Microsoft Word into the RedDot browser-based interface. When the end user hits control-v to paste, the content insertion side-steps RedDot’s built in compliance features. All kinds of nasty Microsoft-specific HTML code can be inserted.

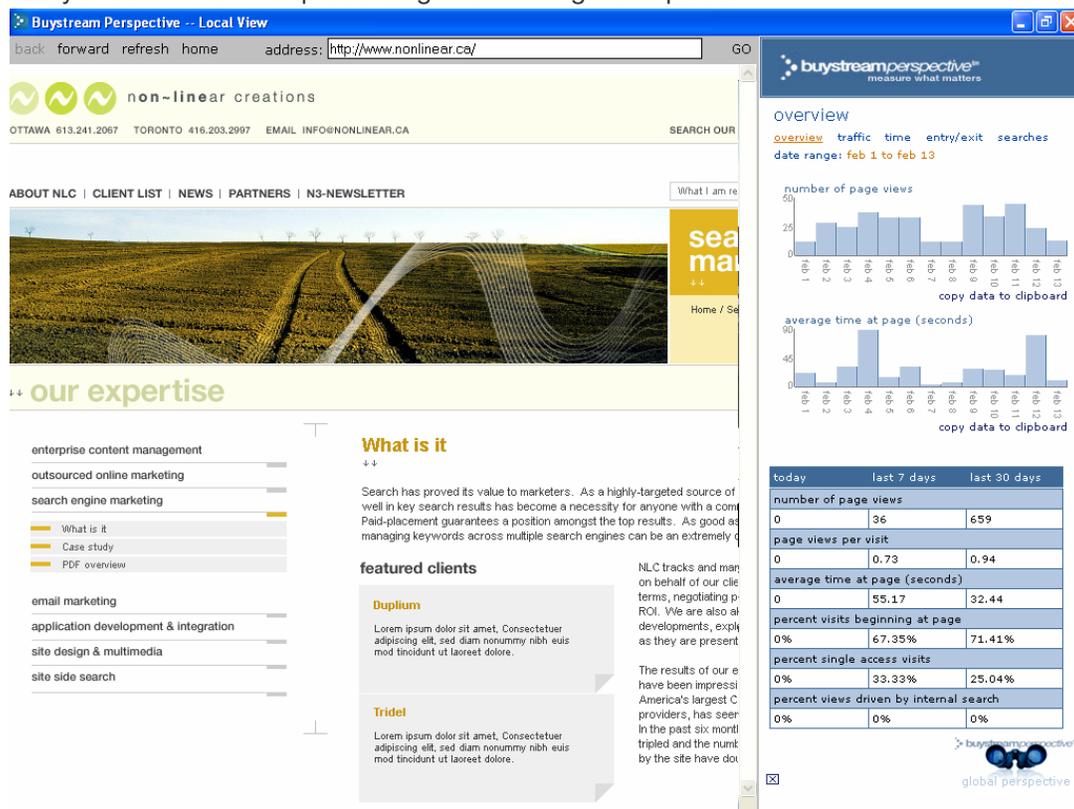
Where compliance with guidelines is mission critical – on a government site, or a site concerned with Section 508 regulations – this “hole” in the RedDot system can be critical.

Fortunately there is a solution. non-linear creations has developed PureCode software that “cleanses” all HTML inserted into the RedDot system. It effectively plugs the hole in the RedDot system and ensures compliance is obtained. You can obtain PureCode directly from non-linear creations by emailing [ninelessons@nonlinear.ca](mailto:ninelessons@nonlinear.ca).

## Tip the Ninth: Perspective is Everything

Once your end users are able to publish their own content to the web site, about one week will pass where everyone is happy and all is good with the world. Then you will receive a phone call asking “is anyone looking at my page?” If you have ever struggled with log analysis tools, you know that is not a fun question to answer. All web analytic software packages will provide you with overall figures – number of visitors to the site, most common paths, etc. – but to find usage statistics about a little know piece of content buried deep in the web site you will need to run a custom report. That can take a long time if you need to reprocess all of the site server logs.

Fortunately, there is a web analytic system designed specifically to integrate with RedDot CMS that solves this problem. It collects information on content use by including a “data tag” inserted into each page of the site. This data tag speaks directly back to the Watchfire AnalyticsXM server – no processing of server logs is required.



The screenshot displays the Buystream Perspective web analytic interface. The main content area shows a website layout with a sidebar menu on the left and a main content area on the right. The sidebar menu includes categories like 'enterprise content management', 'outsourced online marketing', 'search engine marketing', 'email marketing', 'application development & integration', 'site design & multimedia', and 'site side search'. The main content area features a large image of a field with a path, a search bar, and a section titled 'our expertise'. The right-hand side of the interface shows a detailed analytics overview for the date range of Feb 1 to Feb 13. This overview includes two bar charts: 'number of page views' and 'average time at page (seconds)'. Below the charts is a table with the following data:

	today	last 7 days	last 30 days
number of page views	0	36	659
page views per visit	0	0.73	0.94
average time at page (seconds)	0	55.17	32.44
percent visits beginning at page	0%	67.35%	71.41%
percent single access visits	0%	33.33%	25.04%
percent views driven by internal search	0%	0%	0%

Best of all, your end users can get the information they need by themselves. They do not have to call you or anyone else. The product’s “local perspective” allows the end user to navigate the website to find the content of interest. As they navigate, a panel on the right hand side of the screen presents detailed information on visitor interaction with the content they are viewing. If your end user can find the page on the site, they can find the usage information they need: an easy solution to a difficult question.

Email [ninelessons@nonlinear.ca](mailto:ninelessons@nonlinear.ca) for information on this web analytic solution.

## About non-linear creations inc.

non-linear creations (NLC) provides e-business consulting services to an international clientele. Since 1995, we have helped our clients leverage the power of internet technology to achieve tangible business benefits.

NLC has established six areas of practice:

- Enterprise Content Management
- Digital Direct Marketing including Search Engine Marketing and Email Marketing
- Website and multimedia development
- Onsite search optimization
- Custom application development and integration
- Web analytic consultation

Each area of practice is founded on proven methodologies, extensive technology partnerships, and an extensive list of delighted clients.

We have completed more than 500 projects for more than 250 clients including:

- Heinz;
- Mazda;
- Nortel Networks;
- Warner Brothers;
- E\*trade;
- Telus Mobility;
- The World Health Organization (Geneva);
- Alcatel; and
- Drake International.

You can reach non-linear creations:

- by email at [ninelessons@nonlinear.ca](mailto:ninelessons@nonlinear.ca)
- by phone at 613.241.2067
- by fax at 613.241.3086
- by mail at:  
non-linear creations  
987 Wellington  
Ottawa, ON  
Canada  
K1Y 2Y1