Roxen™ User Manual

roxen Platform

www.roxen.com

Introduction

Welcome to Roxen Platform. This manual is intended as a help to someone who uses Roxen SiteBuilder to edit and publish web pages. Users interested in how to write pages that really take advantage of the special features of Roxen Platform, should also read the Web Site Creator manual. Information about managing a Platform server is found in the Administrator's manual. This manual focuses on editing pages in a Platform environment and the particular systems used, such as the Access Control system or how to handle multiple versions of the same document.

Manual structure

This manual consists mainly of three parts. This, the introductory section, will give an overview of the working mechanics and explain the main concepts used when editing files in a SiteBuilder environment. The main part of the manual describes all the moments needed for working with SiteBuilder, with examples and screen shots. Finally, LogView and IntraSeek are described in separate chapters covering the details specific to these subsystems.

Access Control

The access control system in SiteBuilder ensures that users or visitors don't get to see pages that they don't have permission to access. This is done transparently, and also applies to which options are available to the users and editors at any given time.

For visitors, this means that no menu items or search results linking to pages the visitor is not allowed to access will be shown. If a visitor, using IntraSeek, performs a search for any pages containing the word *edit*, he will be shown pages pertaining to text editors or the editors of the site, but he will not be shown links to pages in the content editor, although they are on the site and a user with the right permissions would get to see them.

For users of SiteBuilder, this means that the view of the content editor changes depending on where it is focused. When focusing on a directory where you are permitted to write and edit files, you will have buttons for editing, committing, discarding and so on. When focusing somewhere where you are not allowed to change things - Maybe when looking in the template directory, if you are not working with template programming - and those buttons are not there any more.

Groups and permissions

So, how does this work? Every user is assigned to one or more groups by the system administrator, and every group is then assigned permissions to the different directories of the site.

For instance, you may be a writer and graphic artist, and are therefore assigned to the groups *Everyone*, *Editors* and *Artists*. The first group is a group where everybody is a member, and which has permission to look at only the most public parts of the site, such as the pages that are open to external visitors. The group *Editors* has permission to access the content editor, and to write in the directories that contains the source files for the web pages. *Artists* has permission to write in the directories that contain images.

Your coworker may be a member of the group *Template-Hackers* too, in which case he will also be allowed to write in the directory that contains the site templates. On the other hand, he might not be a member of *Artists*, in which case he might look around in the image directories but are not allowed to make any changes.

So, in effect your group memberships determine where on the site you are allowed to change things, which parts you are allowed to look at but not change and which parts you are not shown at all. As a comparison, an external visitor would probably just be a member of *Everyone* and as such would only be able to reach the public parts of the site. Links to the content editor would not be shown at all.

User management

Most of the management of users and permissions are handled by the system administrators, but there is one thing that the user can handle himself, and that is keeping track of and changing his password. This is done by selecting the tab *Configuration* and clicking *Change password*.

Version control

SiteBuilder allows several users to work on the same site and at the same documents at the same time. This could easily cause problems, when someone edits a document only to find out that in the meantime, someone else has also made changes to the same documents, maybe even in the same places. SiteBuilder avoids this by employing a version control system.

To start off with, SiteBuilder has a central repository where the live documents, the documents that will actually be sent when someone looks at the site, are stored. Together with these files are stored a history of each document - What changes has been made to it, by which user, and when. This enables the system administrator to change a page back to an earlier version of itself by simply undoing the changes made since then, which means that a mistake by a page editor is never likely to be fatal - it is always possible to revert to an earlier, working version. Furthermore, this way SiteBuilder keeps track of who has written each part of a document, and can automatically annotate each document with its authors or log its change history.

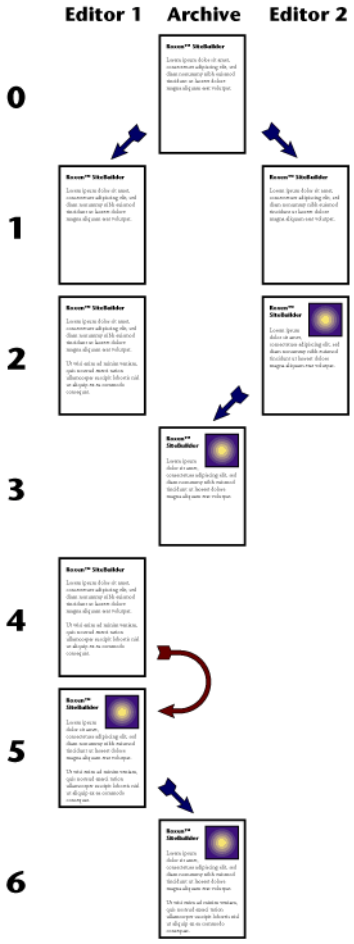
Edit area

It is not possible, or desirable, to let anyone edit the live documents directly. Instead, as soon as you want to edit a document, a copy is made to your personal edit area, where you can edit the document, view it as it would look like on the site, try out different templates, illustrations or whatever. When you are satisfied with the result, you commit your changes to the central repository and the live document changes to reflect your edits.

Conflicts

Now, if somebody else is also editing the same document, it might happen that the document in the central repository has already changed. In this case, SiteBuilder will inform you that the document has changed and that you need to update your edit area to reflect the new changes. When updating, SiteBuilder will attempt to merge the changed document with your edits. Sometimes this will be simple, such as when the edits are in different parts of a long document. However, sometimes there will be a collision, when both you and the other user have been editing the same paragraph. In such a case, both your and the other user's version will be shown, and you'll have to decide which one to keep. Remember, it is always possible to revert to an earlier version. Finally, when you have successfully eliminated all conflicts between the document in the repository and your edits, you may commit your new document to the repository.

Schematic example



Version control

A short example to show how this works:

1 A document is in the central repository, that contains some text. Editor1 and Editor2 starts to edit it and the document is copied to each user's edit area.

2 Editor1 adds some text to the document while Editor2 inserts a picture. Both users work in their own edit area and are therefore not affected by each others changes.

3 Editor2 commits his change, thereby updating the document in the repository. Since no one has changed the document in the repository since Editor2 started editing he has no problems committing.

4 Editor1 tries to commit his change, but this fails since Editor2 has updated the document in the repository.

5 Editor1 uses the *Update* wizard, which updates his document with the changes made since he started editing.

6 Editor1 can now successfully commit his changes to the repository.

Note that the document can be reverted at a later date. Thus the version of the documnet that existed in step three or step one can be recovered.

Templates

One of the features of SiteBuilder is the use of templates to ensure a consistent layout and graphic profile and to assist in having a good navigation system throughout the site. Also, the templates lets the content editor focus on the content he is writing, instead of worrying about HTML and layout.

A template is a file that contains directions for how the page is to be built. It specifies how the headlines on the page will look, where on the page the ordinary text will be placed, whether there should be a navigation system on the page and, if so, how it will look and where it will be placed. In a way, a template may be seen as a mold into which the text is poured: If you select a simple template, the resulting page will be rather simple, while an ornate, complicated template may transform the same text into a page with graphical headlines, menus, a search engine and options to edit the text.

Frameworks and new tags

This is done in two ways. Firstly, the way to make a consistent layout is to have much of the HTML being similar from page to page. If we need a blue bar with a company logo on the top and bottom of every page, then we need to make sure that the code for producing that blue bar is at the beginning and end of every document. The easiest way to do this is to write a framework page, a page which consists of all the necessary elements to make up the layout, and then designate a spot in the framework where the variable content of the page should go.

The other way is to define new tags. Since the Roxen Challenger server allows for using RXML tags, and even defining new tags, we can make the tags do what we want them to do. If we would like an ordinary header, such as `<h1>`, to be rendered as a graphic element using the RXML tag `<gtext>`, then we simply redefine `<h1>` to be a new RXML tag that translates into a command for rendering graphical text. If we want to make sure that no flashing elements appear on the pages, we may redefine the `<blink>` tag into an RXML tag that does absolutely nothing.

Templates in use

Usually all pages of the same kind, such as product presentation pages or intranet document pages, should use the same template. This means that the pages get a consistent layout and feel, of course, but it also means that it will be easy to make site-wide changes to that layout. Adding an extra link to the Christmas Special Offer to every page on the site is as easy as putting it into the framework of the *products* template, instead of having to edit the page for every product the company offers.

Additionally, the template may supply the content writers with extra tags. An intranet template may supply its writers with tags for adding the company logo to the text or tags to mark a word for inclusion in a site wide index page.

Schematic example

The template system works together with the other systems of SiteBuilder. This schematic picture may illustrate the way from content file to the page presented to the user.

1 A content file, consisting of some text and a picture.

2 The content file together with a template. The template added some color and a headline.

3 The next step is to add a navigation menu.

4 Finally, the access control system is invoked, and a few of the menu choices are removed since the requesting user does not have permission to access them.

This is all done internally in SiteBuilder. Now, after all the systems of SiteBuilder has been working on the page, it is ready to be sent to the requesting user.

RXML

Web pages in general is made up of text mixed with HTML tags, sent over the Internet from a web server to a browser

such as Netscape or Internet Explorer. The browser uses the HTML tags to show the page in all its glory to the web-surfing user. There are a lot of different browsers and versions of browser in daily use, and so it is obvious that the set of HTML tags must be rather fixed and agreed upon by all. Despite this, Roxen Challenger offers its own tags, called RXML tags, to extend the sometimes quite limited power of HTML, and it even allows users to define their own tags. How is this possible?

Macro expansion

The answer is that the pages are not sent to the browser as you have written them. Rather, the pages are processed by Roxen Challenger and every instance of an RXML tag is substituted by its HTML equivalent. For instance, the `<tablify>` tag, which takes a list of things and turn them into a rather neat table, is substituted by a rather complicated expression using the ordinary HTML tag `<table>`. The `<gtext>` tag takes text and renders it as an image with selectable fonts, colors, shading, stippling and other graphical effects. When Challenger reaches a `<gtext>` tag, it generates the necessary image and substitutes an ordinary HTML `<img>` tag with a link to the newly generated image.

The same happens with tags defined by the user. Whenever a user-defined tag is encountered, it is substituted with the contents of its definition. This in turn may contain other RXML tags, which is then expanded into their definition, and so on, until it is only ordinary HTML tags left.

So, the RXML tags never reach the browsers. Instead, the server expands every RXML tag in the document to HTML before sending the, by now completely understandable, page to the requesting browser.

Editors

In SiteBuilder, the simplest way to edit a file is to focus on it and press the *Edit* button. Then either a web-based editor will be opened in your web browser or a local editor will be started on your computer. Which editor to use is configured per user.

Web-based editor

The web-based editor has the advantage that it is always available, regardless which computer or operating system are used. It is however lacking in usability, consisting of a text-edit window and nothing more. The web-based editor can only handle text and HTML files.

Local editor

When using a local editor any program on your computer can be used as an editor. Thus each user can choose the program(s) most suitable for her. Any type of file can be edited, for example HTML files, images and Word documents. The local editor is started by a small program, *Roxen Application Launcher*, that must be installed first.

The local editor support must always be initiated from SiteBuilder, by pressing the *Edit* button. From within the editor program *Save* can be used, but not *Save as* nor *Open*. It is possible to save any number of times, until the version control functions of SiteBuilder are used (*Update*, *Commit* or *Discard*). Once one of the version control functions has been used it is only possible to continue editing with a local editor by using the *Edit* button again.

The reason for these limitations is that the link between SiteBuilder and the local editor program is rather fragile. When a local editor is started SiteBuilder creates a special file for it. This file is essentially a copy of the file existing in the user's edit area. This file can only be created by SiteBuilder, and it will be removed as soon as the version control functions are used.

If there are problems with saving a file, maybe because the version control function have been used, there is a remedy. The file can always be saved on the local harddisk and then uploaded to SiteBuilder with the *Upload* button.

Content Editor

Content Editor

The Content Editor is where all actions concerning the site are performed. It consists of four tabs, *Files*, *Access Control*, *Plugins* and *Configuration*. The *Files* tab is for all file related actions, such as creating, editing, downloading and uploading files. The *Access Control* tab is for the access control system, and is only available to users with sufficient permissions. The *Plugins* tab holds extra products, such as LogView and IntraSeek. It will only be shown if such products are installed and if the user has sufficient permissions to access them. The *Configuration* tab handles user customization as well as configuration of the Content Editor.

The Content Editor is reached using a web browser. By default the Content Editor is found on: `http://`*your site*`/admin/`, but this can be customized at each site.

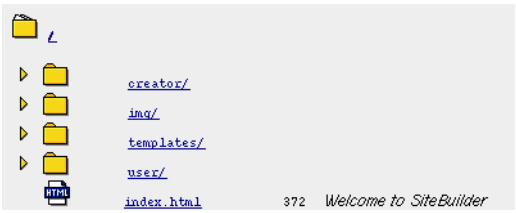
Usually the actual editing of files will be handled by client programs running on the user's computer. They are launched from within SiteBuilder by a helper program, *Roxen Application Launcher*, that must be installed on the user's computer. It is also possible to use FTP to access the files handled by SiteBuilder.

Getting started

Most of the time spent in the Content Editor will be under the *Files* tab. Here is where you edit and enhance the site. The tab consists of two major parts; a row of buttons to the left and the file system to the right, see pictures below.

File system

SiteBuilder stores all files in a version control repository which does not correspond to any file system on your computer. However, this does not make the file system much different from other file systems. The buttons on the left side are used to perform tasks on the directories or files in the file system.



The file system

Focusing

Focusing on a file or directory is done by clicking on its icon or name. Focusing on a file will show information about the file and allows for work on that file. Focusing on a directory shows a file listing of that directory and allows actions that can be performed on that directory. By clicking on the arrow to the left of a directory it is possible to view a directory listing of that directory without focusing on it.

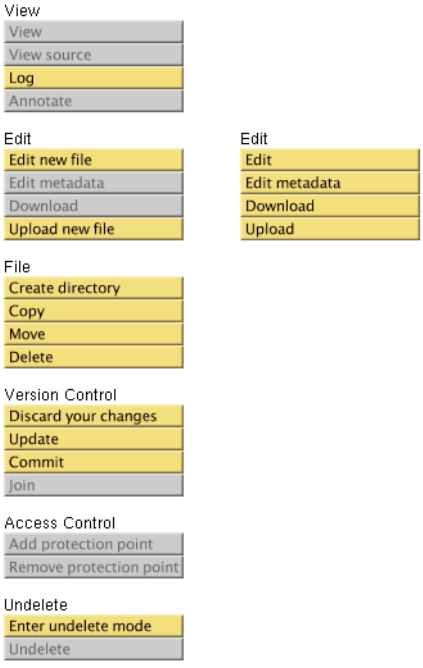
Each file in the directory listings are listed with an icon representing its file type, zero, one or two icons representing the version control status of the file, the file name, the size of the file and finally the file's title. The version control icons are explained further in the Version control chapter. The title of the file is fetched from its meta data.

File types

The type of a file is stored in its meta data. The file type controls how it will be handled by SiteBuilder. A Word document will be handled differently than an image or a HTML file. The file type determines which editor to use when editing the file, how to upload and download the file as well as how the version control system will handle it. It is important that the file type is correct, otherwise it will be impossible for SiteBuilder to handle it correctly.

The file type is related to the file extension, the part of file name after the dot. SiteBuilder does not use the file extension internally, it uses the meta data instead. But when interacting with other systems the file extension will be used as a hint. When uploading a file with a `.doc` extension SiteBuilder will guess it is a Word file. In case the guess is wrong the user has opportunity to correct it.

Buttons



The buttons

As shown in the picture above, some buttons are grayed out, which means that that function is not available for the current file or directory. The buttons are context sensitive, the user's access control permissions, the version control status as well as file type determines whether a button will be grayed out or not. For instance, if the user focuses on a directory in which she is not allowed to change anything, most buttons will be grayed out.

The buttons are ordered in groups according to their area of use:

View

View Views the file, in another window.

View source Views the source of a file, in another window.

Log Shows the version control log of the file or directory.

Annotate Shows the source of the file with annotations about when each line was changed by whom.

Log and annotate are discussed thoroughly in the *Version Control* chapter.

Edit

Edit/Edit new file For a file this button launches the chosen editor for that file type. For a directory it first creates a new file, then launches the editor.

Edit meta data Editor for the meta data of the file, such as its file type, the language it is written in and the template it will use.

Download file Download a copy of the file to the local file system for off-line editing.

Upload file/Upload new file For a file this button is used to uploading a local copy of that file. For a directory a new file will be created from the local file system.

File

Create directory Creates a new directory.

Delete Deletes this file or directory. For files the file will only be deleted in this user's edit area. For directories the directory, together with all files and subdirectories, will be deleted.

Copy Copies a file or directory to another location.

Move Renames a file or directory or moves it to another location.

Version control

Discard your changes Removes any changes made to the file(s) from the user's edit area.

Update Updates the file(s) in the user's edit area with the latest version(s) from the repository.

Commit Commits the file(s) from the user's edit area to the repository.

Join Joins file(s) in two different work areas.

Access control

Add protection point Adds a protection point for this file or directory.

Remove protection point Removes an existing protection point for this file or directory.

Undelete

Enter undelete mode Enters the undelete mode where removed files can be undeleted. Undelete mode is represented by a different background color than normal in the file system and only a few of the function buttons will be available. The background color in undelete mode is pinkish by default.

Undelete Undeletes the file or directory.

Creating files and directories

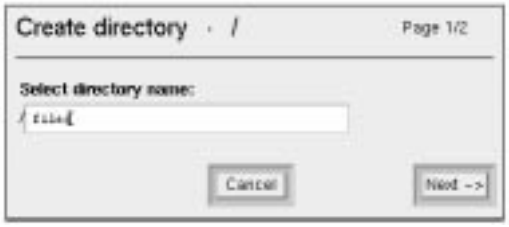
First enter the *Files* tab and focus the directory where the new file or directory should be created. The press the *Edit new file*, *Upload new file* or *Create directory* button to start the wizard that gives guidance through the creation process.

Before creating a new file or directory it is necessary to reflect on the fact that it is not possible to have the same name on files and directories, ever. If a name has been given to a file there can never be a directory with the same name, eventhough the file has been removed, and vice versa. Thus if the directory *index.html/* is created no files named index.html can be created.

Create directory

A directory is created by first focusing on the directory in which the new directory should be created, then pressing the *Create directory* button. It will activate a wizard which will give guidance during the creation process.

In the first part of the wizard the name of the new directory is choosen.



Choose directory name

On the second page of the wizard the choice is confirmed. When the *Ok* button is pressed the new directory will be created.

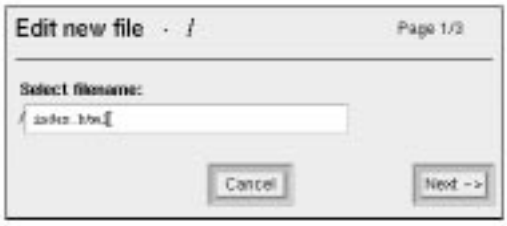
Confirm directory path

Create new files

To create a new file select either *Edit new file* to create a new file or *Upload* to upload an existing file from the local file system. More information about how to upload files can be found in the *Uploading/downloading files* page.

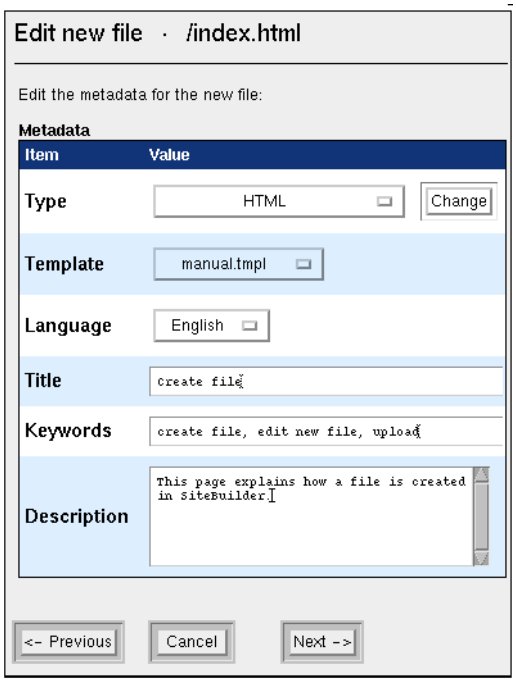
When creating a new file it is necessary to focus on the directory where the new file is to be stored. By pressing the *Edit new file* button a wizard activates to give guidance through the creation process.

The first page in the wizard is used to give the new file a name.



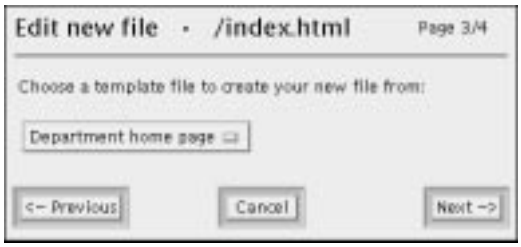
Choose file name

The second page of the wizard is used for editing meta data. It is especially important to enter the correct type for the file, as the file type determines how SiteBuilder will handle the file. For HTML files the template is also important, as it determines what layout will be added to the content.



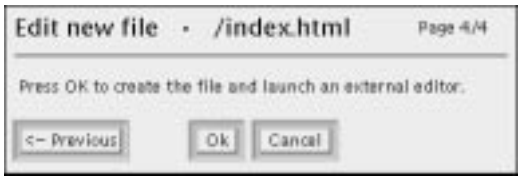
Enter meta data

The third page of the wizard is only available for certain file types. Here it is possible to choose which template file to base the new file on.



Choose template file

The fourth page of the wizard varies depending on file type as well. Either it will ask to launch an external editor or a web-based editor will be shown.



External editor

Web-based editor

Copying/moving files

Copying and moving files and directories is easily done in SiteBuilder. A wizard guides you through the process. The wizard is similar for both operations and is easy to use.

Copy

The first step is to focus on the file or directory that are to be copied and pressing the *Copy* button to launch the wizard that will give guidance through the process.

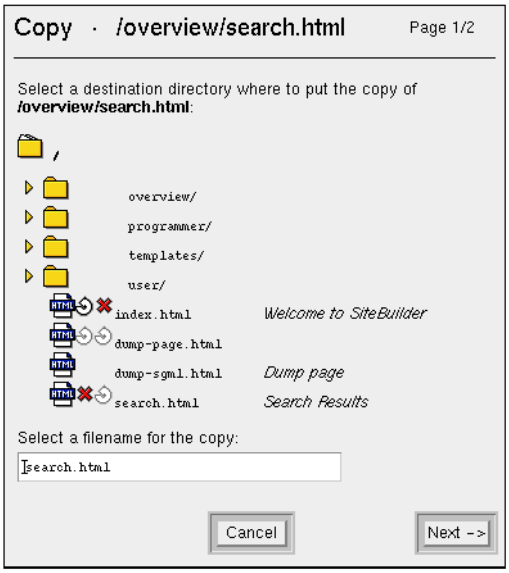
The first page of the wizard is used to find the directory where the copy should be saved, and possibly change the name of the copy. In our example we try to copy the file `/overview/serach.html` to `/user/intraseek/search_result.html`.

At the start we will be viewing the `/overview/` directory. The file system within the wizard is navigated in the same way as the regular SiteBuilder file system, with on exception. It is not possible to click on the directory names, only the icons.

We must choose the *Up to parent directory* link to navigate upwards towards our goal.

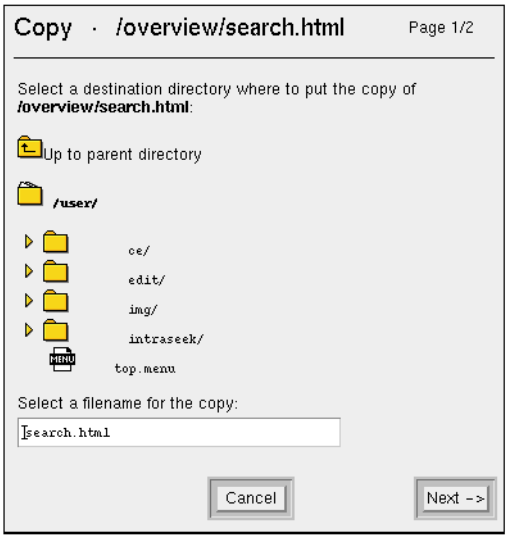
/overview/ directory

After having followed the *Up to parent directory* link we will be looking at the `/` directory, or the root of the file system. From here we must descend the `user/` the directory.

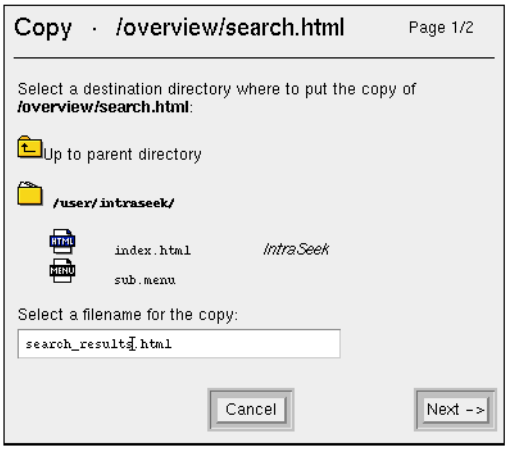


/ directory

Our journey takes us to the `/user/` directory. From here we must descend the `intraseek/` directory.



At the last leg of our journey we are at our destination directory, `/user/intraseek/`. All we have to do before we go to the next step in the wizard is to change the file name to `search_result.html`.

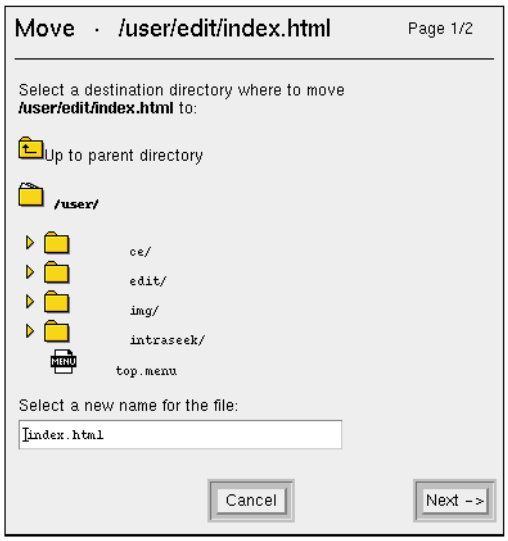


The second page of the wizard asks for confirmation of the operation. By pressing *Ok* we will finish the copying. It will however be necessery to commit the changed file before other users can see the result of the operation.

In case of copying a directory the second page of the wizard will also ask for a log message. As with all directory operations it will be commited to the repository immediately, thus the need for a log message. All files and subdirectories within the directory will be copied as well.

Move or rename

Moving or renaming a file or directory follows the same procedure as copy. Focus on the file or directory that should be moved and the press the *Move* button. Follow the instructions in the wizard and when the wizard is finished commit the changes.



As with the copy wizard it is necessary to confirm the move.

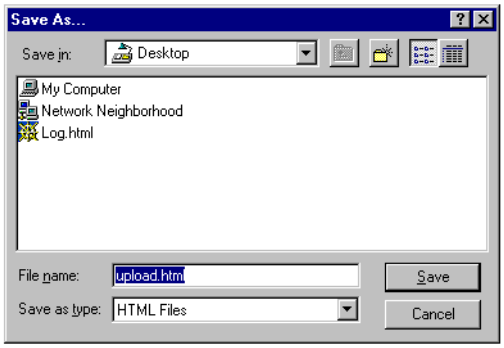
In case of moving a directory the second page of the wizard will also ask for a log message. As with all directory operations it will be commited to the repository immediately, thus the need for a log message. All files and subdirectories within the directory will be moved as well.

Uploading/downloading files

Files are copied between the local file system and SiteBuilder's file system by the *Download* and *Upload* buttons, or by using ftp.

Download

A file is downloaded by focusing on the file and pressing the *Download* button. The file will be copied using the *save file* function of the browser, which allows for browsing the local file system for a directory in which to save the file.



An example download

When a file is downloaded it will be copied to your edit area, so the version control system can keep track of the changes made to the file. This is important if someone else is committing changes to the file while you are editing the content.

The setting *Text download convention* in the *User preferences* controls what line breaks will be used when downloading a text or HTML file. It is important to set this since different operating systems use different line breaks.

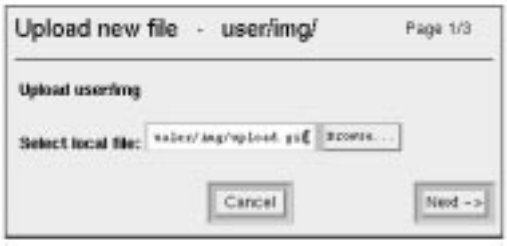
Upload

To upload a file focus on the file and click on the *Upload* button to browse the local file system for the right file to be uploaded.

Upload wizard

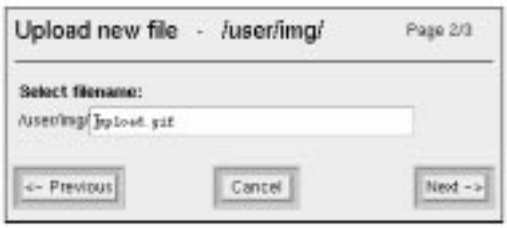
It is important that you are focused on the correct file when uploading. If you are focused on the wrong file you will overwrite that file.

To upload a new file a directory must be focused, not a file. By pressing the *Upload new file* button the Upload file wizard will be activated. The wizard allows for browsing the local file system in search for the correct file.



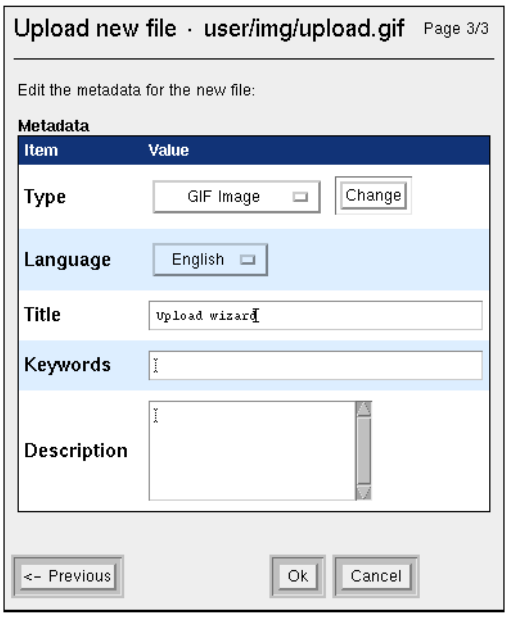
Upload new file wizard

The second page of the wizard gives the possibility of giving the file a more appropriate name than before. As it also gives the directory path to the new file it also acts as a reminder to minimize the possibility of misplacing the file.



Upload new file wizard

The third part of the Upload new file wizard is the Meta data wizard which is discussed in the Editing files chapter.



Meta data wizard

When uploading a text or HTML file it is not necessary to have the right *Text download convention* set, since SiteBuilder will handle all possible line breaks when uploading.

FTP

It is possible to use ftp to upload and download files to SiteBuilder's file system. If, and where, the ftp port can be found is installation dependent.

You usually use ftp together with a graphical client, either a web browser or a special ftp client. When using ftp the topmost directory is usually the different work areas available. This is to make it possible to reach all work areas through ftp.

To use a web browser as a ftp client you type in a ftp URL, usually `ftp://user name@`*your site/*. When using a special ftp client you type in the user name, password and the name of your site separately.

As when downloading through the *Download* button files that you download through ftp will be stored in your edit area. The *Text download convension* will be used to convert the line breaks as well.

It is not possible to do everything through ftp, it is for example not possible to create directories or commit the files through ftp.

Deleting files

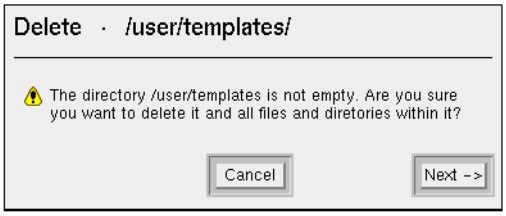
Files

A file is deleted by focusing on it and pressing the *Delete* button. A wizard gives guidance through the deleting process. To finalize the deletion of a file from the file system the file must be committed. Even when a file has been deleted it can be undeleted by entering *Undelete mode* which is described in-depth in the Version Control chapter.

Delete wizard

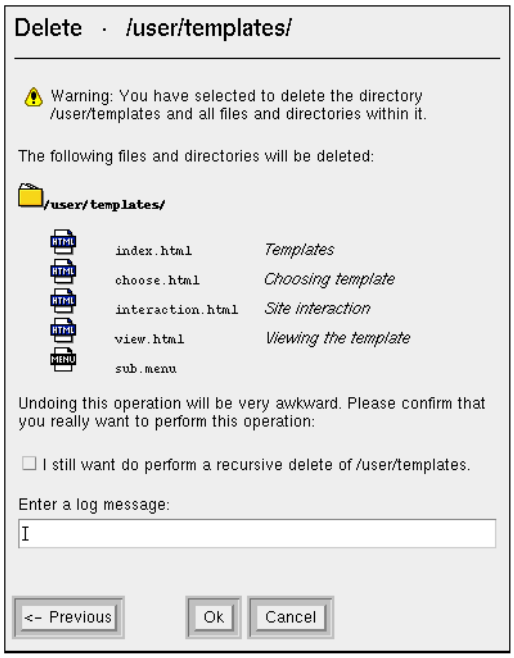
Directories

A directory is deleted by focusing on it and pressing the *Delete* button. All files and subdirectories within the directory will be deleted as well. As with all directory operations it will be commited to the repository immediatly. It will of course be possible to be undelete the directory and any files and subdirectories by entering *Undelete mode*.



The delete directory wizard

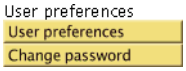
If the directory isn't empty, the wizard will ask if the deletion is to take place.



Confirmation page

Customizing the Content Editor

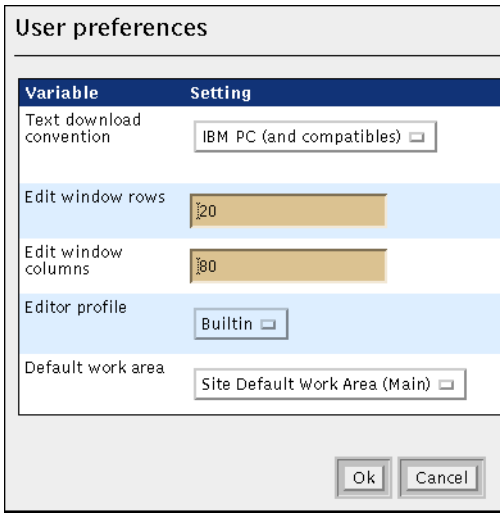
It is possible to personalize the content editor. The settings are found under the *Configuration* tab under the *User preferences* heading.



User preference buttons

User preferences

User preferences are the user's own personalized settings of the content editor.



User preferences

Text download convention Controls what type of line breaks will be used in text and HTML files. Is used when a text or HTML file is downloaded, fetched through ftp or sent to a local editor. It is important to set this setting to be able to work with any editor on the local computer.

Edit window rows The number of rows the builtin web-based editor will use.

Edit window columns The number of columns the builtin web-based editor will use.

Editor profile The editor profile controls which editor will be started, the builtin editor or a local editor. The profile can contain different settings for different file types. The editor profiles are always customized for each SiteBuilder installation, usually to reflect the editor software available at each site.

The profile *Builtin* will always be available. It uses the builtin web-based editor.

Default work area The work area that will be used when the user first connects to the content editor. By default this is *Site Default Work Area*, which is a work area configured by the administrator.

Change password

The *Change password* button allows the user to change his/hers password via a wizard.

Change password

ssword">

Editing files

A file in SiteBuilder consists of the file contents and its associated meta data. The meta data contains the file type, the title of the page, what template should be used as well as information about the page sent to search engines.

Each HTML file is combined with a template before being sent to the user. Usually layout and navigation support are handled entirely by the template, freeing the writer of the content file to concentrate fully on the actual content. It is however still possible to use HTML or RXML tags in the content file, to emphasize text, create diagrams and so forth. Furthermore, the template may provide you with additional tags for formatting, automatic index generation and so on.

Files are usually edited on-line while the user are connected to the content-editor. This is the easiest way, and all functionality of SiteBuilder will be available. It is however also possible to edit files off-line, by first downloading and the uploading the files.

On-line editing is either done through the builtin web-based editor, a local program started by the Roxen Application Launcher or a local program capable of accessing SiteBuilder through ftp. To use a local program is often best since the user then can use the HTML editor, drawing program or word processor she is familiar with.

Local editor

Users using SiteBuilder extensively will most likely use a programs running on their computer to edit files, such as a HTML editors, drawing programs or word processors. When the user press the *Edit* button in the content editor the appropriate program will be started with the correct file loaded.

Built-in editor

Ftp

Files in SiteBuilder can be accessed by FTP if the server has an FTP port configured that is connected to the SiteBuilder. How to configure the FTP port is described in the installation section of the Administrator's manual.

When using FTP, SiteBuilder's file structure appears as a top-level directory containing one sub directory for each Workarea. Inside each Work area directory, the rest of the path is identical to the part that is displayed in the ordin SiteBuilders template system will merge ary SiteBuilder interface. In other words, the FTP path is:

`ftp://`*hostname*`:`*port*`/`*workarea*`/`*documentpath*

The FTP interface does not support access to meta data. When creating new files through FTP, SiteBuilder will make an educated guess as to what kind of file it is, based on its extension, such as .html for HTML files, etc. HTML files will have their title extracted, but not removed, and inserted into the meta data, if SiteBuilder can find a header section containing a title block in the new file.

No version control functions are available through FTP so to commit the file you have to use the Content Editor.

Local editor

Using a local program running on the user's computer is usually the best way to edit files. The user can use the HTML editor, drawing program or word processor she is most familiar with.

By pressing the *Edit* button the appropriate local program is started with the correct file loaded. The user can edit and save normally. In case a HTML file is edited the user will still have to press the *View* button to see the end result after the template has been added. When the result is finished the user has to use the *Commit* button to commit the changes to the version control repository.

It is however very important to close the file in the local editor after commiting. Because it will no longer be possible to save the file, or rather if the file is saved SiteBuilder might not find it. The reason for this is that the file being edited is only available to the local editor while SiteBuilder thinks the user is editing it, or rather while the file exists in the user's edit area. As the user commits the changes SiteBuilder assumes that the user is no longer interested in editing that particular file and removes it from the user's edit area.

In case the user continues to edit a file that has been commited she must save the file on the local hard disk and use the *Upload* button to upload the changes. It is also possible to press the *Edit* button after commiting the changes, in which case SiteBuilder will start the local program again.

The user must always press the *Edit* button in the content editor to start editing a file. It is not possible to use the *Open* function of the local program, since the file will likely not exist in the user's edit area yet. Nor is it possible to rename a file by using the *Save As* function in the local program. SiteBuilder will not find the new file. It is however possible to use *Save As* to save the file to the local hard disk and then use the *Upload new file* button in the content editor to upload the new file to SiteBuilder.

Roxen Application Launcher

To start a local program on the user's computer a small program, *Roxen Application Launcher*, has to be installed. This program is the necessary glue between the web browser and the other programs installed on the user's computer.

SiteBuilder must also be instructed to start the Roxen Application Launcher for the appropriate file types. This is done by choosing the correct *Editor profile* with in *User preferences* wizard under the *Configuration* tab. Which editor profile are available is dependent on the actual SiteBuilder installation. The editor profile controls for which file types Roxen Application Launcher should be started.

What program to start for a particular file type is not configured in SiteBuilder but rather in the registry for the local computer (this assumes the Windows version of the Roxen Application Server is used). This is a security precaution, it must not be possible to start every program on the user's hard disk from the web. Usually there will be a site specific registry file containing the appropriate configurations. It is installed simply by downloading the registry file and double-clicking on it.

Sometimes the web browser will open a dialog about a "possible security hazard", before invoking the Roxen Application Launcher. The user will be asked whether to open the file or save it to disk. It is safe to open it is also safe to instruct the web browser never to ask that question again.

The built-in editor

The builtin web-based editor consists of a simple form that lets the user edit HTML or text files.

To edit a file, focus on the file and select the *Edit* button. When finished editing it's possible to either throwing the changes away by clicking on *Cancel* or selecting *Ok* to save the changes to the user's edit area.

The builtin editor

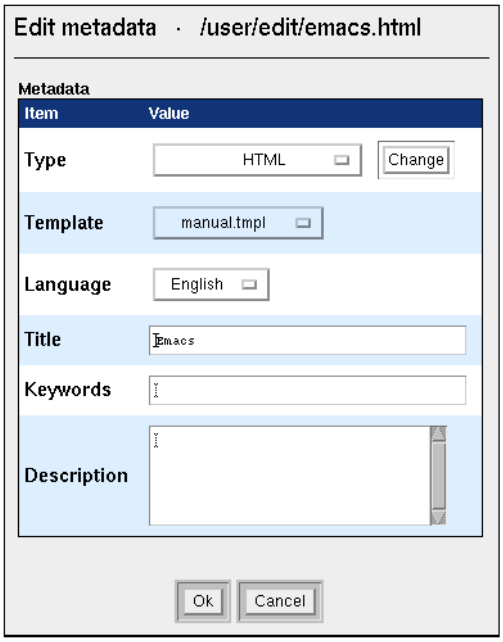
It is possible to alter the width and height of the edit window by selecting the *User preferences* button on the *Configuration* page. It is a good idea to change the size of the edit window to better suit the size of the screen, which in turn makes it easier to work.

By editing a file, the changes are not automatically transferred to the live site. It is always necessary to use the *Commit* button, otherwise no one else will be able to see them.

Meta data

Meta data is information that informs the server how to handle a particular file and it also helps search engines index the file correctly.

In SiteBuilder there is a special meta data editor that helps setting the meta data correct for a particular file. To access the meta data editor, focus on the file and select *Edit meta data*. In the editor the following meta data can be set:



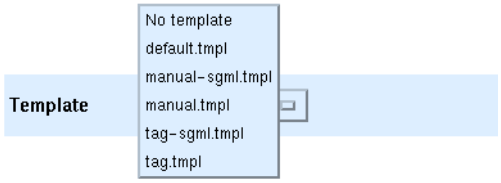
The meta data editor

Type The file type of the file. SiteBuilder treats each file according to its type. It is thus very important to set the correct type. Among other things the file type controls which editor will be used to edit the file and how the version control system will treat the file. Which file types are available can be configured at each site.

Part of the file types list box

Unlike other meta data fields there is a *Change* button after the type select box. To change the type of a file it is necessary to press the *Change* button. This is because the type controls which other meta data fields will be available. The template field will only be available for HTML files.

Template If the file type is set to HTML it becomes possible to chose a template. The template provides the content from the HTML file with layout. Which templates are available depends on the actual site.



The templates available for this manual

Language The language of the file. This is used to determine which character set will be used for HTML and text files. It is important when using languages other than western european languages. Which languages are available depends on the site.

Part of the language list box

Title A title that describes the page more accurately than the file name. The title will be used as the page header and in the navigation interface. This meta data field makes the use of the HTML `<title>` tag unnecessary.

Keywords Document keywords that help search engines index the site better. Use of keywords can give a page better ranking in the search result.

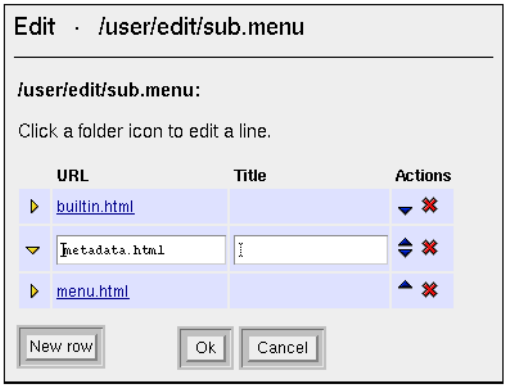
Description A document description that will used in the search result given by search engines. If nothing is entered here the search engine will use a part of the text on the top of the page instead.

The description field can also be used in a navigation interface.

Menu files

Menu files can be used to determine what will be show in the navigation interface. A menu file has the file type *Menu* and might have the extension `.menu`, if extensions are used on the site.

A menu file is edited by focusing on the file and selecting the *Edit* button. A special web-based editor is used. A menu item consists of a URL to the page and a title that should be used in the navigation interface.



The menu editor

After entering the items it is possible to change the order they will be presented on the site by clicking on the arrows. Menu items may also be deleted by selecting the red X to the right of each row. Save the changes of the menu file by clicking on the *Ok* button.

In case the URL is a relative URL to a file on the site it is not necessary to fill in the title field. The title from that file's meta data will be used instead. That way the title need only be changed once, in the meta data, and the change will take effect on every place in the navigation interface.

Note that access control will be used to filter all menu items using relative URLs. That way the no navigation interface will contain links to any pages the current user does not have access to, nor any broken links to pages that does not exist. If the menu file is created before the files in the menu they won't show, until the files have been created.

Only relative URLs are filtered by access control. Absolute URLs will always be shown, regardless of permissions.

Version Control

The version control system allows for full control of all changes made to files in the SiteBuilder file structure. All versions of a file that have been committed to the repository are saved and can be viewed. It is also possible to revert to an earlier version of a file by simply clicking a button.

When files have been modified, one way or another, various symbols will be shown beside the file type symbol. The symbol in the first column represents the user's edit area, while the second represents the file committed to the repository.

The significance of each of these can be interpreted somewhat differently depending on which column the symbol is shown in, and if it is shown in normal or undelete mode. See descriptions below.


The file is created but has not been committed to the repository.


The file is created but has not been committed to the repository, while another user created a file with the same name in the same directory and committed it to the repository.


The file is copied to the user's edit area, but no changes have been saved.


The file has been modified and the changes have been saved in the user' edit area.


Another user has committed changes of the file to the repository. This indicates a possible conflict and an update will be necessary before the file can be committed.


The file is copied to the user's edit area, but no changes have been saved, while another user has committed changes of the file to the repository.


The file has been removed but not committed. After being committed the file will only be seen in undelete mode.


Current user has a copy of the file in the edit area. Another user has removed the file from the work area and has committed the action to the repository.


Current user has saved changes to the file in the edit area. Another user has removed the file from the work area and has committed the action to the repository.

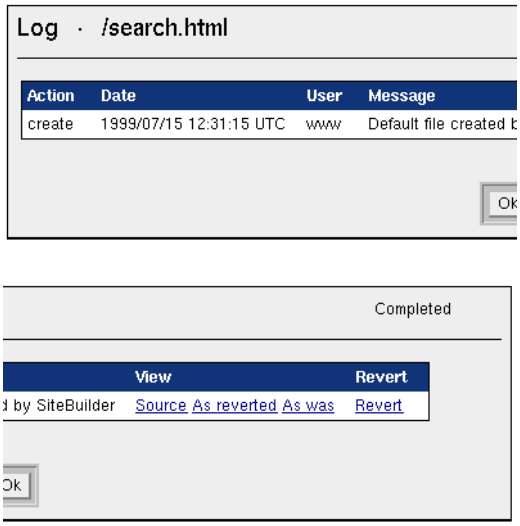

A deleted file, as seen when using the *Undelete mode*.

Log & annotate

The log and annotate wizards helps the users keeping track of all committed versions of the files in SiteBuilder. From the log wizard a file can be reverted to a previous version. The annotate wizard shows which user made a particular change.

Log

Clicking the *Log* button when focusing on a file will show a list of all versions of the file and when and by whom they were committed together with the commit messages. When focusing on a directory the log will show when and by whom the directory was created and perhaps deleted and undeleted.



The log wizard

Each version can be viewed in the following different ways:

Source View the source of that version of the file.

As reverted View that version of the file as it will look if it is reverted. Eventhough an older version of the content file will be used it will be combined with the current version of the template as well as the current version of any images.

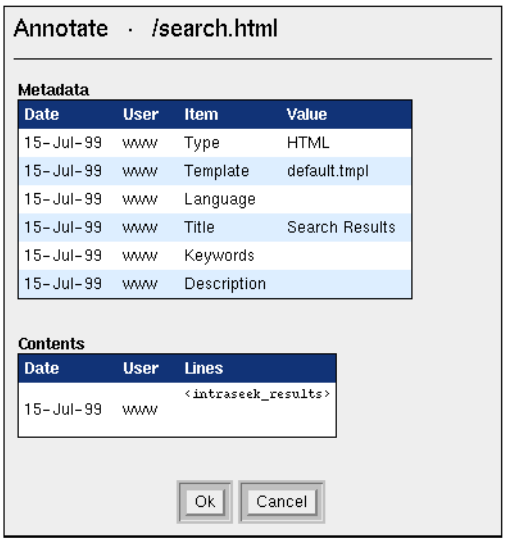
As was View that version of the file exactly as it was. The old version of the content file will be combined with an equally old version the template as well as old versions of any images.

Reverting to a previous version

Choosing *Revert* will replace the present copy in the edit area with the current version from the log wizard. The reverted copy will be stored in the user's edit area and must be commited before other users can see it.

Annotate

The *Annotate* button produces a wizard showing all changes made to the meta data and contents of a file and when and by whom they were committed.



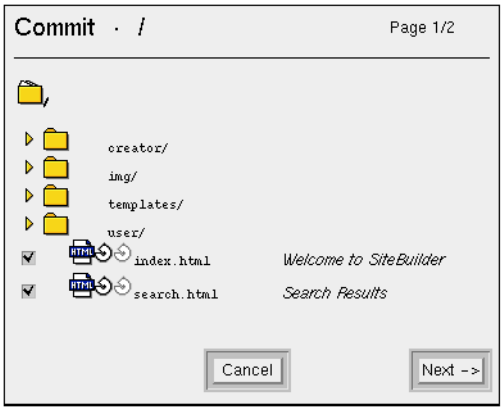
The annotate wizard

Committing files

When creating a new file or making changes to a file available to other SiteBuilder users, the file must be committed. Otherwise, the changes will only exist in the user's private edit area. When committed, the file will be saved in the *repository*.

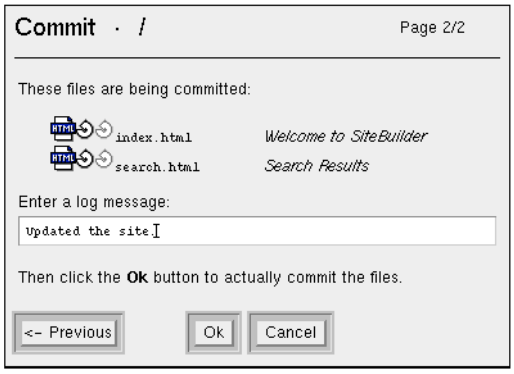
The repository is where a work area stores all files and all changes that have been committed. Even files that have been deleted are stored in the repository.

A file is committed by focusing on the file or the directory containing the file and clicking the *Commit* button.



The commit wizard

Several files can be committed simultaneously by focusing a directory. All changed files in the current directory will automatically be selected. To commit files in subdirectories they must be unfolded, this is done by clicking on the arrow in front of the directory name.



Commit wizard - log message

A log message must be provided to complete the commit wizard. If the messages are written carefully, they will be a valuable help when examining the logs.

If another user has committed changes of a file in the edit area, an *update* must be made to solve possible conflicts before the file can be committed.

Updating files

When a user is editing a file and another user commits changes of that file to the repository, the status icons file will indicate a conflict as shown in figure below.

File indicating conflict

Before the file can be committed it must be updated with the latest version from the repository. While updating a file, a number of conflicts and differences may arise, the different types are described under *Conflicts* and *Differences* below.

Solving conflicts

When focusing on a file it can be updated by clicking on *Update*. When clicking on *Next* the wizard searches for conflicts between the copy in the edit area and the one in the repository.

When focusing a directory, the Update wizard will produce a file listing. After selecting files for update, clicking on *Next* will continue to the *File status summary* which gives the present status of each file selected.

If the wizard finds several files with conflicts, each file has to be resolved separately before they can be updated. To

resolve a conflict, one of the files must be chosen from the list. Files who only contain differences and no conflicts do not need resolving but can be updated directly.

Conflicts

Different content types The content types in the files are different. This conflict must be solved before any of the other conflicts can be handled, because the methods used for solving conflicts depend on the content type of the file. When a content type conflict is resolved, another conflict might arise that is due to something else.

Unresolved conflicts The same part is changed in both files.

Differences

Conflict-free differences For instance:
New data is added.
Old data is removed.
Old data is replaced by new data.

Status

No differences Both files are identical.

Resolved conflicts All conflicts are solved.

Removed from site File or directory is removed from the site.

Actions

Add your file Replace the present version in the repository. The replaced version will still be available in the log due to the version control system.

Discard your file Remove the file from the edit area. All local changes will be lost permanently.

Navigation help

To help the user navigate the wizard, words and different colors have been used to represent different aspects:

Red Represents a conflict and is found in the text to highlight a conflict and to the left in the wizard.

Green Represents the edit area. Text on green background does *only* exist in the edit area.

Blue Represents the site repository. Text on blue background has been added by another user and does *not* exist in the edit area.

Gray Represents old value or content. When defined as old, it is not considered to be a conflict, only a difference between the files.

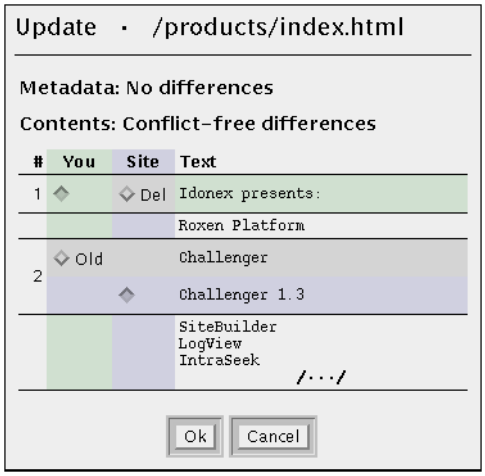
Old The word *Old* is set beside an unselected radio button. It is always represented by gray color.

Del The word *Del* is set beside an unselected radio button. *Del* implies that the data only exists in one file.

Radio buttons A preselected radio button implies that this is the probable choice the user will make. The update wizard always assumes the latest changes should be kept, so these radio buttons are preselected.

The radio button should be selected in front of data to be kept and the *Del* button in front of data to be removed. Data with only an unselected radio button and no *Del* button in front will be deleted.

By selecting a file, the wizard shows the conflict resolving mode. Here, the changes in a file necessary to make an update are made by choosing what data should be kept and what should be removed.

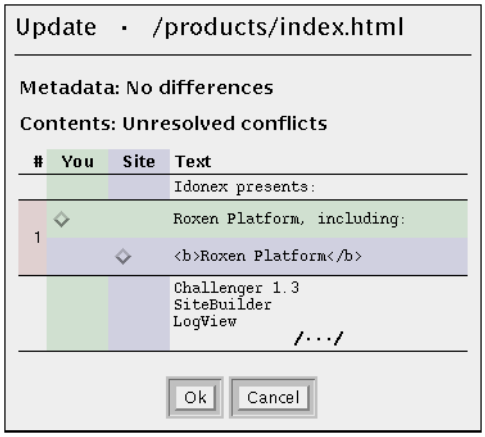


Conflict-free update example

The image above shows an update situation with only conflict-free differences.

1 Text added by current user.

2 Text changed in the repository.



Update example showing conflict

This example shows a conflict where two users have changed the same part of the content. The radio button in front of the text should be selected for the text to be kept. When the proper radio buttons are selected, choose *Ok* to update the file.

After solving

If several files need to be updated and not all files are conflict free, an update is still possible since the conflict free files will be updated. When all updates are finished, the files can be committed.

Aborting changes

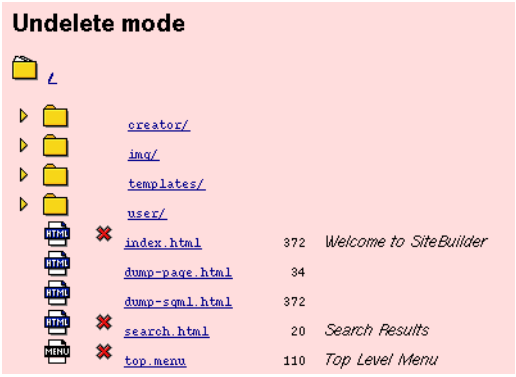
Changes made to a file since the file was last committed can be canceled by choosing *Discard your changes* from the menu.

The discard wizard

If a file is changed the wizard will ask for confirmation before removing the copy from the edit area.

Restoring deleted files

A file that has been committed to the repository will be stored forever and if deleted it can be restored, or *undeleted*. Deleted files can be seen in the file system after clicking the *Enter undelete mode* button.



Undelete mode

When focusing on a deleted file the *View* options and the *Undelete* button will be available. All versions of the file can be viewed and restored. Clicking on the *Undelete* button will restore the focused file.

Undeleted file

The undeleted file must then be committed, which is only possible after leaving undelete mode, before other users will be able to see it.

Files that cannot be restored

To be able to restore a deleted file it must be stored in the repository of the version control system. A file that was deleted before being committed consequently will not be found when entering undelete mode.

Work areas

This chapter focuses on how to handle work areas. The concept of using multiple work areas comes from needing several copies of a live site functioning as advanced test versions of the live site. This makes it possible for the user and designer of the site to enter and evaluate content and layout before publishing it on the live site, without being afraid of causing damage.

Only a part of the work area functionality is explained here because the functionality is divided between the administrator and the user. The configuration of a work area is left to the administrator. The part that considers the user is the Work Area select box in the header and *Join* under the *Files* tab.

Changing work area

Often two work areas are used. One containing the live site and the other containing a test version. Changing between work areas is necessary to see the results when, for example, the live site is updated with data from a test version. The work area select box, shown below, can be found on top of the page in the Content Editor. By selecting a work area and pressing the *Ok* button, you easily can navigate between work areas.

Work area select box

Changing work area is done by selecting the proper work area from the select box. The choice must be confirmed by clicking on *Ok*. Forgetting to click *Ok* can have unpleasant consequences. Since the work areas often are similar, it is not too difficult to do changes in the wrong work area.

Joining work areas

Using two work areas is useful when you want to work completely separate from the live site and have even better control over all changes on the site before launching it onto the world wide web. When the site is ready to be launched all the data must be transferred to the live site. This is done by using the *Join* button under the *Files* tab.

By joining two work areas you merge the two versions of the **same** file or directory from each work area with each other. You cannot join files that have copies in your edit area, they must be committed first.

Solving problems

The wizard begins by asking you to choose which work area to join your focused file or directory with. By clicking on *Next* the wizard searches for conflicts in those files. The different types of conflicts and differences you may encounter are described below:

Conflicts

Different content types The content types in the files are different. This conflict must be solved before any of the other conflicts can be handled, because the methods used for solving conflicts depend on the content type of the file. When a content type conflict is resolved a new conflict might arise that is due to something else.

Unresolved conflicts The same part is changed in both files.

Differences

Conflict-free differences For instance:

- New data is added.
- Old data is removed.
- Old data is replaced by new data.

Doesn't exist in *work area* File exists in one work area but not in the other.

Status

No differences Both files are identical.

Resolved conflicts All conflicts are solved.

Actions

Add to *work area* Select to add the file or directory to *work area*.

Remove from *work area* Select to remove the file or directory from *work area*.

If the wizard finds several files with conflicts, each has to be resolved before continuing to the final step in the wizard. To

resolve a conflict, click on the arrow to the left of the file name.

Joining several files

The image above shows examples of different problems that may occur when joining several files from the two different work areas *Main* and *Test*".

Navigation help

To help the user navigate the wizard, words and different colors have been used to represent different aspects:

Red Represents a conflict. It is found in the text, to highlight a conflict and to the left in the wizard.

Green Represents the repository of your current work area.

Blue Represents the repository of the work area you have chosen to join with.

Gray Represents old settings or data. When defined as old it is not considered to be a conflict, only a difference between the files, the data has been changed in one file but not in the other.

Old The word *Old* is set beside an unselected radio button. It is always represented by gray color.

Del The word *Del* is set beside an unselected radio button. *Del* implies that the data only exists in one file.

Radio buttons A preselected radio button implies that this is the probable choice a user will make. Select the radio button in front of data you want to keep and the *Del* button in front of data you want to remove. Data with only an unselected radio button and no *Del* button in front will be deleted.

By selecting radio buttons the conflicts will be resolved and data removed or added to the work area. The handling of conflicts and differences between files is very similar to the update procedure. The update page contains a few examples.

After solving

To complete the join operation, all conflicts must be resolved in the join wizard. When this is done, click on *Next* to enter the final step in the wizard. Enter a log/commit message to complete the operation, then click on *Ok* to finish the operation.

Before selecting *Ok* to finish the operation it is very important that you are sure of what you are doing, and that all choices in the wizard have been made correctly. After all, joining work areas is an important operation and it is hard to undo the operation, especially if you have joined several files. To undo a join operation you have to manually revert each file or delete the unwanted files or directories. To revert a file, focus on it and select the *Log* button under the *Files* tab.

LogView

This chapter contains information on different ways of using LogView to view the extracted statistics.

Using LogView

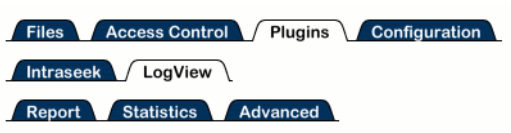
LogView lets the user view the gathered statistics in an ordinary web browser. If LogView has been installed on the Roxen virtual server with the URL

`http://hostname/intranet/`

the default URL to the LogView configurations page is

`http://hostname/intranet/logview/`

However, the administrator may choose another location within the virtual file system. The configurations page can also be found under the *Plugins/LogView* tab in SiteBuilder.



The Plugins/LogView tab

If the administrator has specified that access to the LogView pages should be password protected, the user will get a pop-up dialog where he has to log in before he can access the LogView page.

On the top of all the LogView pages is located a button menu which helps the user to navigate between the four pages *Statistics, Report and Advanced*. These pages provide different ways of looking at the gathered statistics.

The main difference between the *Statistics, Reports* and *Advanced* pages lies in what possibilities the user has to affect the presentation. The *Reports* page contains a number of reports that has been prepared by the administrator, and is therefore the most static one - all the user needs to do is to select which report he wishes to see. The *Statistics* page is the easy way to create reports, and provides basic possibilities to control what statistics to see and in which form, and in the *Advanced* page the user has even more freedom to control the details.

These three levels of control possibilities makes it possible to provide quick-and-easy access for the users who either does not need full control or can have the administrator or site creator tailor a report for them, while the users that really need full control over the details can still get it.

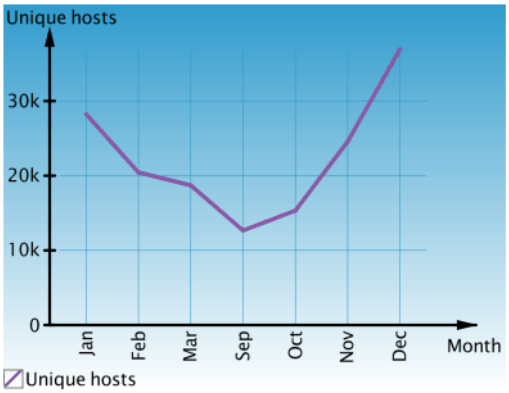
Logging sites

When someone retrieves a page from a web site, all the actions this person does is saved to a file in a special log format. Every action from downloading files, fetching HTML-pages with pictures to what web-browser or operating system the user has is saved as plain text in this file. This file can later be analyzed to find out various behaviors the site's users have.

The process of building a log file works something like this: the users browser sends requests to your Web server for various resources. These resources may include HTML, graphics, audio, and whatever other types of files are on the site. Upon receiving the request from a browser, your Web server accesses the file and sends it back to the client browser.

A lot of useful information is buried in this file. Information about where bottlenecks and errors occur in the server or on the site or where the most popular features on the site are.

The information gathered in the log files can as all statistics information be made to fool those not as attentive or those who don't have any knowledge about the area the statistics is for. Here is a screen-shot that there is obviously something fishy about.



It is easy too fool those not paying attention

To make the best use of statistics information it is important to know what all functions and special terms concerning LogView means. Hence, a compilation of all the important terms and functions in LogView can be found in the *Terms and functions* page.

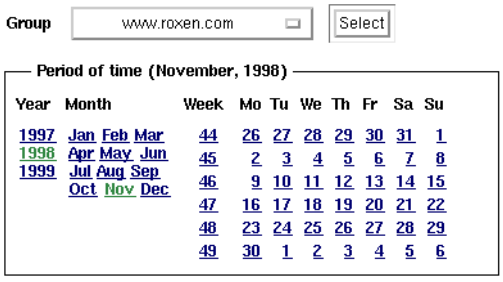
Statistics

The statistics page provides an easily overviewed and almost self-explanatory way of looking at statistics gathered by all the functions that are enabled on the current statistics group.

Selecting statistics group

In the field labeled *Group* the current statistics group is shown. If several groups are defined, the user can select one of them in a list box. By default only one group is defined. It takes its data from the log of the Roxen virtual server that LogView is installed on, and the name of this group is the same as the name of the server.

If no hits have been received since the creating of the selected statistics group, this is shown by the message "No items are logged". This should be remedied within one hour after the site has been accessed.



Select group and period of time

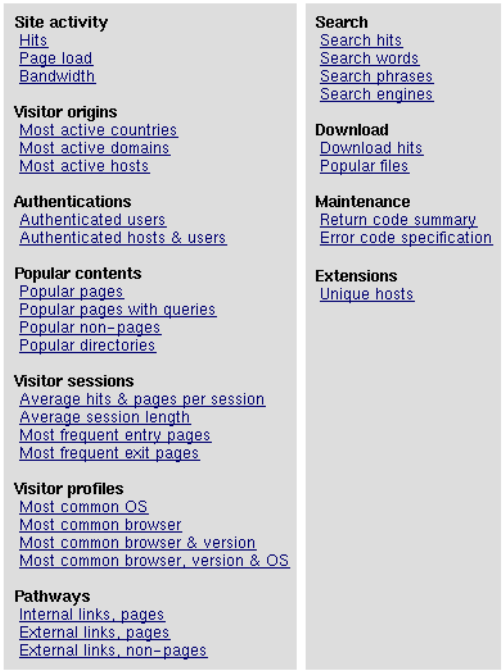
Selecting period of time

In the *Period of time* box, the period of time for which to show statistics can be specified. The currently selected period of time is displayed within parentheses right next to the label on the top of the box. The highlighted periods inside the box are the ones for which logged data is available, and clicking one of them simply selects that period of time. By default only years and months are shown, but once a month is selected the box is expanded to also contain weeks and days.

On the *Statistics* page, statistics can only be shown for a single year, month, week or day. To specify a more exact time interval, the user has to go to the *Advanced* page.

Selecting kind of statistics

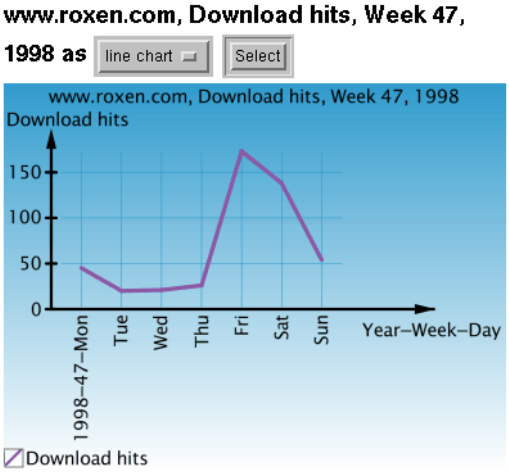
Below the *Period of time* box, a number of available functions are listed in groups. To see statistics of a certain kind the user just has to select the corresponding function in the list. An overview over the functions can be found in the *Terms and functions* page.



Select statistics function

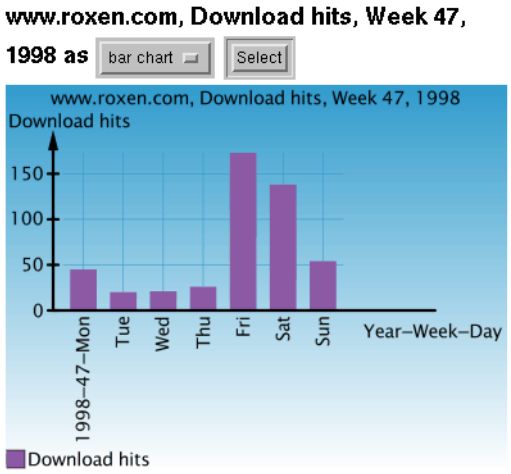
Selecting format

Once the user has selected the kind of statistics to show, a report is displayed at the bottom of the page. This report is in a default format that is dependent of the kind of statistics that was selected. If the user is not pleased with this format, he/she can choose another one in the list box in the title of the report. Below is a selection of statistics reports for www.roxen.com under week 47, 1998. These reports selected by first choosing the www.roxen.com group, then clicking on 1998, Nov and last week 47.



Download hits - line chart

This report is selected by clicking on the function *Download hits* and then choosing *line chart* from the pull down menu.

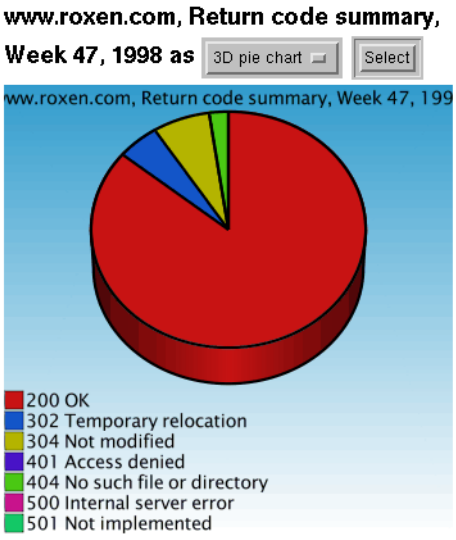


Download hits - bar chart

This report is selected by clicking on the function *Download hits* and then choosing *bar chart* from the pull down menu.

External links, pages - table

This report is selected by clicking on the function *External links, pages* and then choosing *table* from the pull down menu.



Return code summary - 3D pie chart

This report is selected by clicking on the function *Return code summary* and then choosing *3d pie chart* from the pull down menu.

Reports

The report page is used for viewing reports that has been prepared by the LogView administrator.

Advanced

In this page, the user can control all of LogView's capabilities for viewing statistics.

The advanced statistics page

Selecting group

Select statistics group

As was the case on the *Statistics* page, the statistics group, if more than one is available, can be selected in a list box.

Selecting report

Select type of report

The user can select the kind of report from all the kinds that are available for the selected statistics group in the first list box on the row.

The list box after *as* default labeled *Day* is used to specify the granularity of the diagram or table. If *days* is selected here, a resulting bar chart will have one bar for each day for

which data is available, and a table will accordingly have one row for each day. If instead *month* is selected, the bar chart will have one bar for each month, with a height that is the the sum of the all the values of the days of this month.

The list box labeled *per* selects what should be printed for example as labels on the bars of a bar chart. To get a chart where each day has it's own bar and the label Year-Week-Day (e g 1998-25-Tue) the user can select the report to be shown "as day, per week". A report that is "as day, per month", has the label Year-Month-Day (e g 1998-Jun-16) on its bars instead.

Trying to set the *as* value larger than the *per* value gives the following error message:

```
LogView Error: unit cannot be greater than per
```

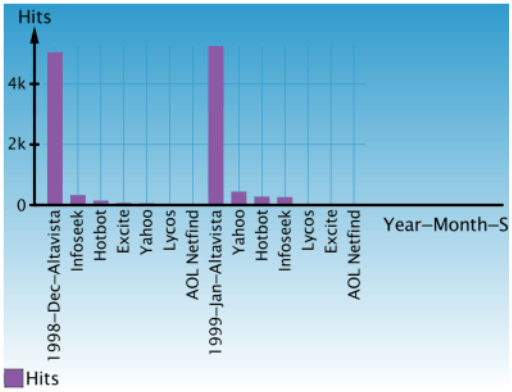
Setting both values equal has an interesting effect when using the Sum feature, as we will see in the next section.

Selecting presentation

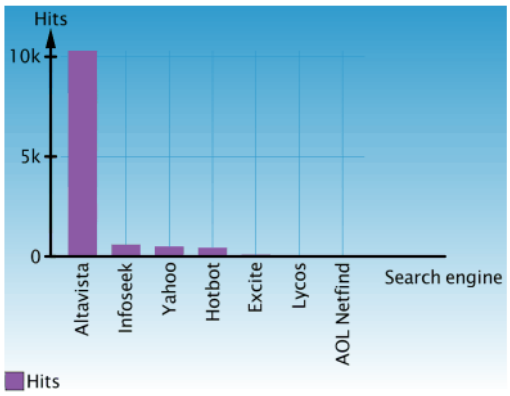
Select presentation

In the first list box, the user can choose between *Append* or *Sum. Append* is the default, and gives sums for each day, week, month or year. Using the *Sum* feature changes this behavior somewhat - instead of summing the data inside each day, month or year the data for several such units can be summed, to answer questions like "how many hits were processed in total between the 1998-06-02 and 1998-07-11?" However, to use this feature we have to use the trick to set the *as* and *per* time periods the same.

The two pictures below shows the difference in *Append* and *Sum* when the setting is month per month. The report is *Search engines* and the time period is between December, 1998 and January, 1999.



Append



Sum

Another scenario is when the user has set the report to days per week, and the time period is three weeks, where *Append* will present it as three weeks in a row. Sum will instead present it as one week, where each weekday is the sum of that weekday in all three weeks. Due to lack of space no screen-shots of this scenario is possible in the manual.

There is also a possibility to set the maximum number of rows to be shown. The term rows seem to imply the use of a table, but the limitation works on all kinds of diagrams as well, and then limits the number of bars, pie slices or other corresponding items.

Selecting from and to values

Select time period(TU)

The user can specify the exact time period for which statistics should be shown by selecting the years, months and days for the beginning and end of the period.

Selecting display

Select display type

When the user has specified all of the above, he can start the display process by pressing one of the display buttons in the row at the bottom of the page. These are divided into categories by their form of presentation.

Tables:

- HTML table

#	Year	Month	Hits
1	1998	Sep	469 396
2	1998	Oct	548 271
3	1998	Nov	787 083
4	1998	Dec	933 046
5	1999	Jan	693 880
6	1999	Feb	738 378
7	1999	Mar	770 851
8	1999	Apr	676 930
9	1999	May	632 297
10	1999	Jun	785 343
11	1999	Jul	610 257

Hits at www.roxen.com, September 1998 through July 1999

Settings: Hits as Month per Year, Append

- ASCII table

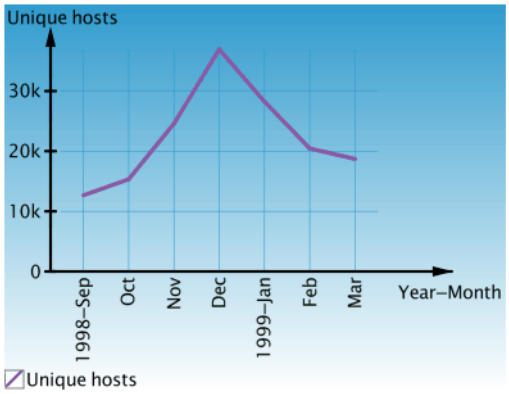
#	Domain	Pages	Hits
1	unknown	196459	1271793
2	aol.com	20943	206164
3	liu.se	28682	194564
4	ne.jp	11034	96744
5	telia.com	10273	92497
6	home.com	11431	91798
7	swipnet.se	9303	89153
8	t-online.de	10559	88394
9	uu.net	13339	88301
10	ac.at	40203	82999
11	co.uk	9681	78069
12	algonet.se	17114	72942
13	mxs.nl	5718	72617
14	co.jp	7658	51506
15	progmet.com	8037	51372
16	or.jp	7186	44082
17	com.au	9074	36041
18	com.br	3423	35499
19	ac.uk	3619	33242
20	rr.com	5217	33204

Most active domains at www.roxen.com, September 1998 through July 1999

Settings: Most active domains as Month per Year, Sum

Charts:

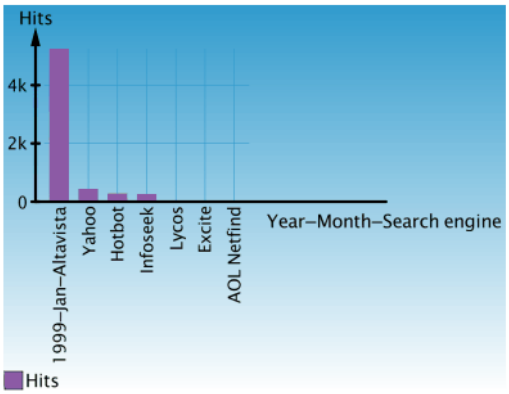
- Line chart



Unique hosts at www.roxen.com, September 1998 through March 1999

Setting: Unique hosts as Month per Year, Append

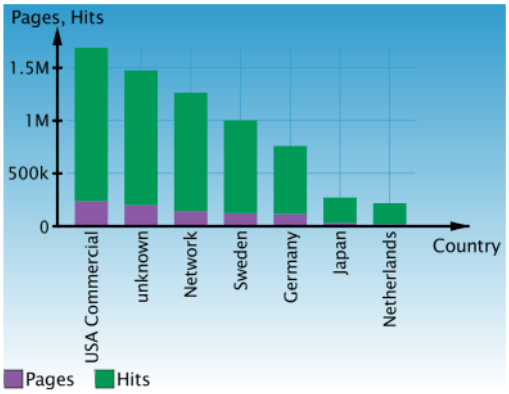
- Bar chart



Search engines at www.roxen.com, January 1999

Settings: Search engines as Year per Month, Append

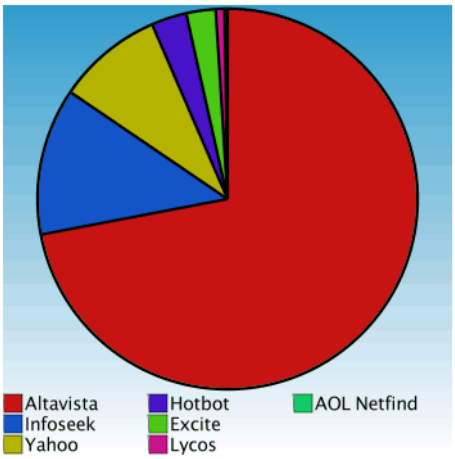
- Sum bars chart



Most active countries at www.roxen.com, September 1998 through July 1999

Settings: Most active countries as Month per Month, Sum, max 7 rows

- 2D pie chart

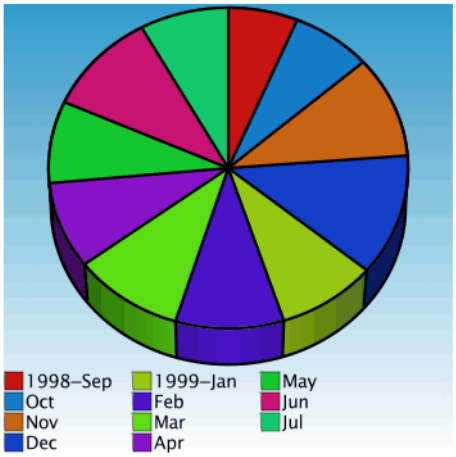


Search engines at www.roxen.com, January 1999 through July

1999

Settings: Search engines as Month per Month, Sum

- 3D pie chart



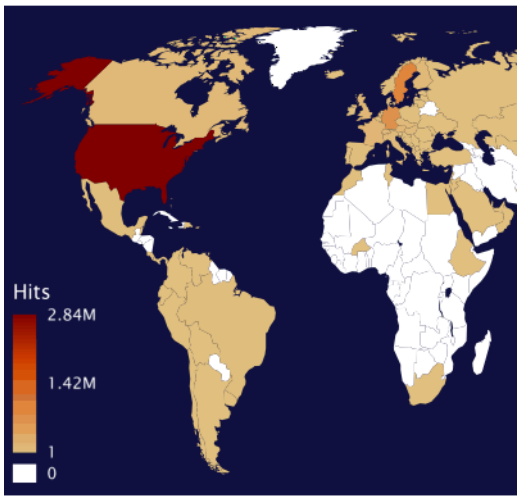
Hits at www.roxen.com, September 1998 through July 1999

Settings: Hits as Month per Month, Append.

Miscellaneous:

- Map

A world map, nice for displaying statistics over Most active *countries*. Not available in any other case. Below is a slightly cut version of the map, made for this manual.



Most active countries at www.roxen.com, September 1998 through July 1999

Settings: Most active countries as Month per Year, Append

- Export

Exports the table in a tab/newline separated format, suitable for MS Excel.

- Tag

Displays the LogView RXML tag with all the parameters (see appendix B) that generates the created report. Great for cut-and-paste!

```
<logview  group='www.roxen.com'
 report='most active countries'
 unit=month  per=year
 op=sum  max=100
 from-year=1998
 from-month=9
 from-day=1
 to-year=1999
 to-month=7
 to-day=31
 display=world-map>
```

Most active countries at www.roxen.com, September 1998 through July 1999

Settings: UTSL! (Use the source Luke!) or read the tag-code and find out the right settings.

Terms and functions

Terms

This part describes the difficult words and terms in LogView. These words are either web analysis terms or used by LogView to describe it's functions.

Bandwidth In a general sense, this term describes information-carrying capacity. In LogView this applies to how much data (in bits per second) that has been transferred from the server.

Hit A hit is a request from a browser to the server. Every element of a requested page, text, pictures, and interactive items is counted as one hit to the server. A normal web page contains an average of at least six elements, i.e. six hits. Hits is not the preferred way of measuring traffic on a site as the number of hits often varies widely.

Page load Page load aka page views or page deliveries gives the number of times a web page has been requested. Page load, not hits, are the preferred counting method site-traffic estimates and measurement.

Search engine The search functions can identify the following search engines: AOL Netfind, Alta Vista, Excite, Infoseek, Hotbot, Lycos and Yahoo.

Session A session is tied to a hostname where the hits has occured up to 20 minutes between each other.

Period of time/time period In LogView reports can be created to show the data between two points of time. What is in between those two points of time is the period of time.

User variable Variables which the administrator or the privileged user can set in one of the LogView wizards found in the configurations interface. Some of the functions makes use of these variables when monitoring the log files. This term is only used when describing some of the functions below.

Functions
This part describes all the functions used to present data about the site in LogView.

Site activity
This group of functions monitors the overall activity on the site.

Hits Gives a summary of the number of hits during the period of time.

Page load Summaries the number of hits per time period and page.

Bandwidth Calculates the average number of sent bits per second and time period.

Visitor origins
This group of functions monitors where hits originates from.

Most active countries This function describes which countries are generating the most hits.

Most active domains This function tells which internet domains are responsible for generating most hits. A domain can be roxen.com, foobar.net, etc.

Most active hosts This term gives a summary of each hosts hits. The most active host is the one who has the most hits. A host can e.g., be an IP-number.

Authentications
Gives information about user logins.

Authenticated users Gives a summary of the number of hits/time period and authenticated user.

Authenticated hosts & users Gives a summary of the number of hits/time period, hostname and the authenticated user.

Popular contents
This group of function monitors what has been requested on the site.

Popular pages Gives a summary of hits/time period and page.

Popular non-pages Gives a summary of hits/time period and non-page, where non-pages are all files which are not pages e.g., a zip-file or a pdf-document.

Popular directories Gives a summary of hits/time period and directory.

Visitor sessions
This group of functions gives information about visitors coherent sessions.

Average hits & pages per session This function calculates the average number of hits and the average number of pages per time period and session.

Average session length This functions calculates the average session length in minutes per TU.

Most frequent entry pages Summaries the number of hits per time period and page where the page is the first page accessed in a session.

Most frequent exit pages Summaries the number of hits per time period and page where the page is the last page accessed in a session.

Visitor profiles
This group of functions gives miscellaneous information about the users.

Most common OS Summaries the number of hits per operating system and time period.

Most common browser Summaries the number of hits per browser and time period.

Most common browser & version Summaries the number of hits per unique browser version and time period.

Most common browser, version & OS Summaries the number of hits per unique browser version, it's operating system and time period.

Pathways
This group of functions monitors links that points towards the site.

Internal links, pages Summaries the number of hits to the linked page (the "to" page) where the link-page (the "from" page) was an internal page. Pages that match the user variables *Server Domain Name* or *Server Alias* are counted as internal pages e.g., all pages on *your.site.com* are internal pages.

External links, pages Summaries the number of hits to the linked page where the link page was an external page.

External links, non-pages Calculates the number of hits to files which are not pages e.g., zip-files, pdf documents, etc.

Search
This group of functions gives information about visitors who hahave found the site using search engines.

Search hits Summaries the number of hits where the previous page was a search engine.

Search words Summaries the number of words in the search string per time period where the previous page was a

search engine. The stopwords 'and', 'or', 'not', 'the', 'an', 'a', 'is' and 'are' are not counted.

Search phrases Summaries the number of phrases per time period where the previous page was a search engine and the phrase a search string.

Search engines Summaries the number of hits per search engine where the previous page was a search engine.

Download

This group of functions monitors which downloads have been made.

Download hits Summaries the number of hits per time period where the filename matches the uservaiable *Download files* i.e., total number of downloads.

Popular files Summaries the number of hits per time period and filename, where the filename matches the user variable *Download files* i.e., sorted by files.

Maintenance

This group of functions gives information necassary for troubleshooting.

Return code summary Summaries the number of returned status codes.

Error code specification Summaries the number of returned status codes per time period and filename, where the status code is an error code i.e., separated from 200 and the interval 300-399.

Extensions

This function monitors the number of unique hosts who have accessed the site.

Unique hosts Summaries the number of hits per time period and unique host, where a host can be represented by a hostname or an IP-number. It should be noted that both hostname and IP-number can represent the same host, which implies that they will be counted as two different hosts. This is a known bug.

IntraSeek

How to search with IntraSeek

This text can be added to a web page to tell the users how to use the IntraSeek search form.

It is possible to use quotation marks (?) and asterisks (*) to broaden searches. A search for "net*" might match "netscape", "nethack", "network" and so on. A search for "int??net" matches "intranet" as well as "internet". Note that IntraSeek requires that the user specifies at least three characters in front of the "*" notation, and that there is no difference between lower- and uppercase searches.

You can use quotation marks to search for a phrase. For example, a search for "John Carl Smith" will search for persons with this name. Without quotes, you would have get any pages that use any of those common names.

Boolean

Boolean search allows you to include or exclude documents containing certain words through the use of the operators AND, OR, NOT and XOR.

If you just type some keywords, the "OR" operator is assumed. For example, searching for "coca-cola pepsi jolt" will accept all documents containing one or more of any of these three words.

Searching for "coca-cola AND pepsi" only displays the documents that contain BOTH coca-cola and pepsi.

Searching for "(coca-cola AND pepsi) AND NOT jolt" will display all documents that contain the words "coca-cola" and "pepsi", but will leave out documents containing anything about "jolt".

You can add any level of "(" and ")" to build evaluation trees.

Searching for "(coca-cola XOR pepsi)" will display all documents that contain the word "coca-cola" or the word "pepsi", but not both.