Site Studio Publishing Utility Administration Guide
10g Release 3 (10.1.3.3.0)

March 2007
Contributing Authors: Will Harris

Contributors: Sean Cearley

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.
# Table of Contents

## Chapter 1: INTRODUCTION
- Guidelines for publishing ............................................... 5
- Guidelines for customizing Site Studio pages ......................... 6
- Documentation ............................................................... 7
- Conventions used in this guide ........................................... 8

## Chapter 2: GETTING STARTED
- Creating a new publish destination ...................................... 9
- Publishing a Site Studio web site ........................................ 10
- Error handling for publishing ............................................ 12
- Customizing error handling ............................................... 13
- Standard HTTP error codes .............................................. 14

## Chapter 3: PUBLISHING UTILITY ADMINISTRATION
- Current status .............................................................. 15
- Add website ................................................................. 16
- All destinations ............................................................ 17
- Create new destinations .................................................. 17
- Event logs ................................................................. 18
- Create event log .......................................................... 19
- Logging facilities .......................................................... 20
- View log messages ....................................................... 21
- General settings .......................................................... 21
Table of Contents

Chapter 4: CONFIGURATION (XML) ELEMENTS

Base elements ................................................................. 25
Filters .............................................................................. 29
filterset element (child of job element) ......................... 30
filter element (child of filterset element) ...................... 32
authentication element (child of filterset element) ........... 35
Triggers ............................................................................ 35
cmd triggers ................................................................. 35
http-post triggers .......................................................... 36
Java triggers ................................................................. 37
Programming Java triggers ............................................. 37
Sample ........................................................................... 38
Sinks .............................................................................. 38
DTD for job and filter elements ........................................ 39

Appendix A: Configuration file properties

The root element ................................................................. 41
The options element .......................................................... 41
The timeFormat element .................................................... 45
The log element ................................................................. 46
The proxy element ............................................................ 48
The ssl element ................................................................. 49
The services element ........................................................ 50
The admin-server attribute .............................................. 50
The content-server element .............................................. 51
The ice-server element ...................................................... 52
The master-server element .............................................. 53
The database element ...................................................... 54
The content-sources element .......................................... 55
The extensions element .................................................... 56
The factory element ......................................................... 57
The ldap element ............................................................ 57
The j2ee element .............................................................. 58
## Table of Contents

- The knet element .............................................. 60
- The license-server host element ................................ 60
- The knet-server host element .................................... 60
- The soap-server host element ................................... 60
- The tracking-server host element ..................... 61
- The clickthru-server element ............................ 61

## Index
CHAPTER 1

Introduction

This administration guide describes how to use Site Studio Publishing Utility to publish a Site Studio web site. With Site Studio, you store the web sites you build in Content Server. At some point, you may find that you would like to publish the site from a Content Server to a web server that is not running a Content Server instance. This process is referred to as publishing.

The Publishing Utility allows you to publish a Site Studio web site from a Content Server environment to a pure web server environment (that is, one running Microsoft IIS, Apache, etc.).

The Publishing Utility creates a static snapshot of a dynamic site by traversing all the links in a web site (visiting all of the linked pages) and downloading a copy of each page and all of the resources (images, flash movies, etc.) on each page. Your entire web site (including the content of queries, layout pages, fragments, contributor data files, and native documents) will be copied and published to the new server.

In this section:
- Guidelines for publishing
- Guidelines for customizing Site Studio pages
- Documentation
- Conventions used in this guide

Guidelines for publishing

To publish a web site from a Content Server environment to a web server, all the content of your site will be copied by the Publishing Utility. However, the copy of the site must be both complete and self-contained. That is, the copy must contain all of the resources of the original site, and pages in the copy must refer only to resources within the copy itself. To create a self-contained copy, the Publishing Utility rewrites any absolute (full) URLs within the original site to relative URLs to allow the static copy to be hosted on a different hosting instance.

The Publishing Utility uses the following procedures to identify and evaluate links and resources contained in a particular page:
- When handling HTML, the Publishing Utility looks for any attributes in tags that may contain links (for example, the ‘HREF’ attribute in an ‘A’ tag or the ‘SRC’ attribute in an ‘IMG’ tag), and downloads these images. If the link is absolute, the link in the copied site is updated to refer to the downloaded image (via a relative URL) instead of the original image. See “Guidelines for customizing Site Studio pages” on page 6 for more information.
- When handling JavaScript, the Publishing Utility looks for specific JavaScript commands that are commonly used to load a new image or change what is being displayed on the page. When those specific commands are encountered, it evaluates the JavaScript and interprets each command. However, an unrecognized JavaScript command may cause the Publishing Utility to not include a referenced page or fail to download an image. See “Using JavaScript in links for published pages” on page 6 for more information.

Note: See the Site Studio documentation for more information.
Guidelines for customizing Site Studio pages

When customizing Site Studio–generated JavaScript or writing your own components, be sure to use relative URLs for all images, movies, etc., that are referenced in the script; avoid concatenating strings to build up paths to resources. This will make it easier for the Publishing Utility to properly identify and evaluate the external references.

Pay special attention to those items included in a dynamic list or using JavaScript patterns, and also to naming conventions for published pages.

Using JavaScript in links for published pages

You may want to specify a JavaScript pattern in a link to be used in your published web pages. Certain patterns must be used for the Publishing Utility to identify and evaluate JavaScript references and links in order for these references to be converted to relative references in the published version.

The Publishing Utility looks for specific patterns in the embedded JavaScript of web pages. When these patterns are encountered, the Publishing Utility evaluates the JavaScript and interprets the command.

- The Publishing Utility evaluates general JavaScript patterns that are commonly used to load a new image or change what is being displayed on the page. See “General JavaScript patterns” on page 6 for more information.
- The Publishing Utility also evaluates certain Site Studio–specific JavaScript patterns. See “Site studio-specific JavaScript patterns” on page 6 for more information.

Note: An unrecognized JavaScript command may cause the Publishing Utility to not include a referenced page or fail to download an image.

In all patterns listed below, double quotation marks ("") and single quotation marks (') are treated as equivalent. All data in bold are simply examples; the crawler will attempt to match any valid URL within the quotes.

General JavaScript patterns

<table>
<thead>
<tr>
<th>JavaScript fragment</th>
<th>Link to be traversed</th>
<th>Replacement</th>
</tr>
</thead>
<tbody>
<tr>
<td>something.src = &quot;foo.gif&quot;;</td>
<td>foo.gif</td>
<td>none</td>
</tr>
<tr>
<td>top.location.href = &quot;other.html&quot;;</td>
<td>other.html</td>
<td>none</td>
</tr>
<tr>
<td>foo.open(&quot;test.html&quot;);</td>
<td>test.html</td>
<td>none</td>
</tr>
<tr>
<td>frame.replace(&quot;something.html&quot;);</td>
<td>something.html</td>
<td>none</td>
</tr>
</tbody>
</table>

Site studio-specific JavaScript patterns

<table>
<thead>
<tr>
<th>JavaScript fragment</th>
<th>Link to be traversed</th>
<th>Replacement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Var g_httpCgiUrl = &quot;...&quot;;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>link(XXX)</td>
<td>?IdcService=SS_GET_PAGE&amp;ssDocName=XX /static/doc_XXX</td>
<td></td>
</tr>
<tr>
<td>nodelink(YYY)</td>
<td>?IdcService=SS_GET_PAGE&amp;nodeId=YYY</td>
<td>See note 1</td>
</tr>
</tbody>
</table>
Notes:
1. A hierarchical path is constructed using the URL Page Name and URL Directory Name properties of the site and the nodes. If no properties are defined the node label with underscores is used.
2. If the locations of these images have been altered, their names have been changed, or new assets have been added, the locations will not be picked up in the crawling process.
3. The parameters of these JavaScript calls, which are references to files or directories, are converted to a relative reference in the published version.

Important: Do not customize Site Studio pages by appending URL parameters to a node reference, as any appended parameters are ignored (i.e., any node parameters not defined using Site Studio will not be used in naming the pages). For example, a URL appended after nodeId=YYY is ignored.

Documentation

The documentation for Site Studio Publishing Utility is available in the installation directory, as online Help, and on the web site.

<table>
<thead>
<tr>
<th>Documentation</th>
<th>Format</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release Notes</td>
<td>PDF</td>
<td>Available on the DVD or in the distribution ZIP file.</td>
</tr>
<tr>
<td>Installation Guide</td>
<td>PDF</td>
<td>Available on the DVD or in the distribution ZIP file.</td>
</tr>
<tr>
<td>Administration Guide</td>
<td>PDF,</td>
<td>Available on the DVD or in the distribution ZIP file.</td>
</tr>
</tbody>
</table>
Conventions used in this guide

The User Guide contains the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bold</strong></td>
<td>Indicates an item that you select in the Designer or Contributor interface, such as a button or menu, in order to perform a specific task: Click <strong>OK</strong> to confirm the deletion.</td>
</tr>
<tr>
<td>&gt;</td>
<td>Indicates a menu choice. For example, “Choose <strong>File &gt; Open</strong>” means “Click the <strong>File</strong> menu, and then click <strong>Open</strong>.”</td>
</tr>
<tr>
<td><strong>Code</strong></td>
<td>Indicates the actual code used by Designer and code you can enter in Source view in a layout page.</td>
</tr>
</tbody>
</table>
CHAPTER 2

Getting started

Site Studio Publishing Utility allows you to publish a web site from a Content Server environment to a pure web server environment (that is, one running Microsoft IIS, Apache, etc.). All of the content of your site, including content picked up by queries, will be copied and published to the new server.

The Publishing Utility is the server side of the system and either a Subscription Client or an FTP server is the client side of the system. The Subscription Client is generally installed on every machine that hosts a web server (unless you choose to use an FTP server).

Note: The Subscription Client is used as the client side component for both Connection Server and Site Studio Publishing Utility.

This is an overview of the installation and setup:

- Install the Publishing Utility.
- Install a Subscription Client instance (or use an FTP server).
- Create a new publish destination (configure the Publishing Utility to communicate with the Subscription Client or FTP server).
- Configure the Subscription Client to communicate with the Publishing Utility (or use an FTP server).

Note: For the steps to install and configure the Publishing Utility and Subscription Client, see the Site Studio Publishing Utility Installation Guide (installation-guide.pdf).

In this section:

- Creating a new publish destination
- Publishing a Site Studio web site
- Error handling for publishing

Creating a new publish destination

On the Publishing Utility administration interface, click Destinations and then Create New Destination to define your destination details, destination type, and Subscription Client or FTP server information.

To create a new publish destination:

1. Enter a Destination Name. Use a descriptive name for the publish destination (for example, PublicWebServer or ProductionServer7).
2. For Description, you can enter additional descriptive information (optional).
3. For Destination Type, select one of these options:
   - Subscription Client: The content is delivered to a Subscription Client instance. You must define Subscription Client Details below.
   - FTP Server: The content is delivered directly to a designated FTP server. You must define FTP Server Details below.
4. Enter a Destination Login and Password for the destination (Subscription Client or FTP server). The login and password are used by the Subscription Client to authenticate itself to the Publishing Utility.

5. If your destination type is a Subscription Client, enter the Destination Push URL. This is the URL of the hosting server where the Site Studio web site is to be published (e.g., http://productionserver7:8886 or http://106.106.106.1:8886).

   If this is a new Subscription Client installation, and you have not already done so, you must configure the Subscription Client to communicate with the Publishing Utility. For more information, see the Installation Guide for Site Studio Publishing Utility (installation-guide.pdf).

6. If your destination type is an FTP server, enter this information:
   - URL of the FTP server
   - FTP server port number
   - FTP server login name
   - FTP server password
   - A subdirectory to place content in

7. Click Save.

### Publishing a Site Studio web site

On the Publishing Utility administration interface, click Add Website to define your site details, update schedule, and delivery information.

**To define your site details:**

1. Enter the user and password to access the Site Studio web site.
   
   The user name and password entered here defines what user account is used to access the Site Studio web site; and therefore controls whether you will be able to publish a site without getting access control errors.

2. Enter the Server CGI URL.
   
   This is the URL to the instance of Content Server that is hosting your Site Studio web sites (e.g., http://server7/stellent/idcplg).

3. Click Connect.
   
   The Site ID drop-down list is automatically populated.

4. Choose a Site ID from the list.
   
   The Site ID is the unique identifier of your Site Studio web site in the Content Server.

5. Click Generate Manifest URL.
   
   The Manifest URL field is automatically populated.

6. Enter a descriptive Site Name (e.g., “Business Website” or “ProductionServer7”).
7. For Site Description, you can enter additional descriptive information (optional).

8. Select your delivery option:
   - **Incremental:** Content will be delivered incrementally: that is, if delivery of a package of content items is interrupted, the content items successfully delivered and retrieved so far will not be discarded. Delivery of the remaining items will resume at the next delivery attempt, as dictated by the subscription's delivery rule. Use this option if delivering to an FTP server.
   - **Atomic:** For each user, the Publishing Utility must successfully deliver all of the items sent in each delivery; otherwise, no items will be delivered. (No incremental delivery.) Only available when delivering to a Subscription Client.
   - **Synchronized:** The Publishing Utility must successfully deliver all the items sent in each delivery to all users; otherwise, no items will be delivered to any users. The synchronized delivery option is useful when you need to reliably distribute identical copies of content, in parallel, to multiple servers (e.g., to an array of mirrored web servers that support load-balancing or failover). Only available when delivering to a Subscription Client.

To define your update schedule:

1. Choose an update option:
   - **Every day:** Select this option to check the content every day of the year.
   - **Only on:** Select this option and then select the desired day(s) of the week to check content only on those days.
   - **Only on these dates:** Select this option to check content every month on certain dates. You can type days of the month as numbers separated by spaces. Use “last” to specify the last day of the month. (For example, entering 10 20 last sets checking for the 10th, 20th, and the last day of the month.)
   - **Manual Update:** Select this option to prevent the site from being checked. With this option, no content will be automatically published. This option is useful if you want to manually check and publish the site (by clicking Update Now on the Status page).

2. Enter a **First update at** value to specify the start time for checking to occur.
   Type a time in hour and minutes and select AM or PM (for example, 3:40 PM).

3. Enter a **Update every** value to specify the interval between checks.
   Type a number and select Hours or Minutes from the list (for example, to check once a day, enter 24 and choose Hours).

4. Enter a **Last update at** value to specify the final time for checking to occur.
   Type a time as hour and minutes and select AM or PM (for example, 3:40 PM).

To select your delivery destination:

1. Choose which destination to deliver to by checking the subscribe option for that destination.
2. Click **Save** when you are done defining your site details, update schedule, and delivery information.

### Error handling for publishing

Administrative control over error handling is provided by pre-set configuration variables. These settings define whether a returned HTTP response is handled as a hard-error or soft-error indication, or simply ignored.

**Note:** The terms “hard-error” and “soft-error” are simply descriptive terms to indicate the relative seriousness of a returned HTTP response. By default, certain HTTP responses are evaluated as hard-errors, while other HTTP responses are evaluated as soft-errors. However, these defaults can be customized.

#### Understanding error handling

When there is no error, the contents of the Site Studio web site is published to the publish destination. If an HTTP response is detected, an error code is returned. The Publishing Utility configuration settings define how these errors are handled.

In most cases, the default settings will be adequate for most publishing needs. However, by editing the configuration values you can change the settings. This provides an additional level of customization and control over error handling.

- Network IO Errors (physical network problems, like an inability to connect) are always treated as hard-errors.
- The allowed threshold for hard-errors and soft-errors can be customized by changing the values in the hard-error threshold and soft-error threshold entries.
- HTTP responses in the 500 range are evaluated as hard-errors by default. This can be customized by listing them explicitly in the soft-errors or ignore-errors list.
- HTTP responses in the 400 range are evaluated as soft-errors by default. This can be customized by listing them explicitly in the hard-errors or ignore-errors list.
- HTTP responses in the 300 - 100 range are ignored by default. This can be customized by listing them explicitly in the soft-errors or hard-errors list.

**Note:** Soft-error indication is an essential part of the interface, since it may not always be possible to access particular information from the content server.

**Tech Tip:** If you encounter errors in the crawl, the recommended course of action is to determine why the errors are occurring and try to resolve them rather than to immediately increase your threshold. In many cases, they will be simple errors such as missing images or broken links. These are easy to fix and involve simple corrections.

#### Configuration settings for error handling

<table>
<thead>
<tr>
<th>Configuration Setting</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>hard-error-threshold</td>
<td>Defines the number of hard errors allowed. If the defined number is exceeded, publishing will fail.</td>
<td>1</td>
</tr>
<tr>
<td>soft-error-threshold</td>
<td>Defines the number of soft errors allowed. If the defined number is exceeded, publishing will fail.</td>
<td>5</td>
</tr>
</tbody>
</table>
Customizing error handling

Publishing error control is configured by an element in the options tag of the sitestudio.config file (located in the Publishing Utility installation directory). These options affect all offers on the server.

**Note:** This configuration file may not exist after a clean install of the Publishing Utility, but will be generated when you create your first Site Studio offer. The server needs to be shut down for changes to be made to this configuration file.

By editing the configuration settings you can control whether the Publishing Utility evaluates a returned HTTP error as a hard error or soft error indication or define how many hard or soft errors will be allowed.

For example:

- By defining the hard-error threshold as 5, more than five hard errors will prevent publishing of the site.
- By defining the soft-error threshold as 9, more than nine soft errors will prevent publishing of the site.
- By adding 404 and 405 to the hard-errors entry, these error codes will be evaluated as hard errors along with the 500 range of codes (the 500 range of codes are evaluated as hard errors by default).
- By adding 501 and 503 to the soft-errors entry, these error codes will be evaluated as soft errors along with the 400 range of codes (the 400 range of codes are evaluated as soft errors by default).
- By adding 509 and 405 to the ignore-errors entry, this error code will be ignored (not evaluated as either a hard or soft error).

<table>
<thead>
<tr>
<th>Configuration Setting</th>
<th>Description</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>hard-errors</strong></td>
<td>This allows you to specify individual error codes that will be treated as hard errors. There are no wildcards allowed in this list, you need to list each error code explicitly (separated by a comma).</td>
<td>The 500 range of error codes are evaluated as hard errors by default.</td>
</tr>
<tr>
<td><strong>soft-errors</strong></td>
<td>This allows you to specify individual error codes that will be treated as soft errors. There are no wildcards allowed in this list, you need to list each error code explicitly (separated by a comma).</td>
<td>The 400 range of error codes are evaluated as soft errors by default.</td>
</tr>
<tr>
<td><strong>ignore-errors</strong></td>
<td>This allows you to specify individual error codes that will be ignored (not evaluated as either hard or soft errors). There are no wildcards allowed in this list, you need to list each error code explicitly (separated by a comma). <strong>Note:</strong> Use caution when adding error codes to this list.</td>
<td>The 100–300 range of error codes are ignored by default.</td>
</tr>
</tbody>
</table>
Example of modified sitestudio.config file:

```xml
<options>
    <error-control
        hard-error-threshold="5"
        soft-error-threshold="9"
        hard-errors="403, 404"
        soft-errors="501, 503"
        ignore-errors="509" />
</options>
```

## Standard HTTP error codes

This section lists the standard error codes that are part of the Hypertext Transfer Protocol (HTTP). For more information on the standard HTTP error codes visit the World Wide Web consortium (W3C) website:

http://www.w3.org/Protocols/

<table>
<thead>
<tr>
<th>HTTP Protocol Status Codes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 Continue</td>
<td>403 Forbidden</td>
</tr>
<tr>
<td>101 Switching Protocols</td>
<td>404 Not Found</td>
</tr>
<tr>
<td>200 OK</td>
<td>405 Method Not Allowed</td>
</tr>
<tr>
<td>201 Created</td>
<td>406 Not Acceptable</td>
</tr>
<tr>
<td>202 Accepted</td>
<td>407 Proxy Authentication Required</td>
</tr>
<tr>
<td>203 Non-Authoritative Information</td>
<td>408 Request Time-Out</td>
</tr>
<tr>
<td>204 No Content</td>
<td>409 Conflict</td>
</tr>
<tr>
<td>205 Reset Content</td>
<td>410 Gone</td>
</tr>
<tr>
<td>206 Partial Content</td>
<td>411 Length Required</td>
</tr>
<tr>
<td>300 Multiple Choices</td>
<td>412 Precondition Failed</td>
</tr>
<tr>
<td>301 Moved Permanently</td>
<td>413 Request Entity Too Large</td>
</tr>
<tr>
<td>302 Moved Temporarily</td>
<td>414 Request-URL Too Large</td>
</tr>
<tr>
<td>303 See Other</td>
<td>415 Unsupported Media Type</td>
</tr>
<tr>
<td>304 Not Modified</td>
<td>500 Server Error</td>
</tr>
<tr>
<td>305 Use Proxy</td>
<td>501 Not Implemented</td>
</tr>
<tr>
<td>400 Bad Request</td>
<td>502 Bad Gateway</td>
</tr>
<tr>
<td>401 Unauthorized</td>
<td>503 Out of Resources</td>
</tr>
<tr>
<td>402 Payment Required</td>
<td>505 HTTP Version not supported</td>
</tr>
</tbody>
</table>
CHAPTER 3

Publishing Utility administration

This section provides information on using the Publishing Utility administration interface. You can view the Help system in a web browser such as Internet Explorer or Netscape Navigator, and you can use conventional web browser controls to navigate and view the Help topics (Back, Forward, Refresh, etc.).

Click the Help icon to view the online Help. Each dialog box has a Help button that takes you directly to a topic that provides descriptive information on the functionality of that dialog box.

From each Help topic, click the Show Navigation button in the navigation bar to view the complete Administration Guide online.

In this section:
- Current status
- Add website
- All destinations
- Create new destinations
- Event logs
- Create event log
- Logging facilities
- View log messages
- General settings

Current status

This page allows you to view the current status of all web sites and provides publishing details.

Site Name: The site name. Click to view Site Details for this web site displays.

Job Schedule: The job schedule set (e.g., “Manual Update”).

Last Update: The day and time of the last update.

Job State: The job state (e.g., “Completed”) and the number of errors and warnings is listed. Click to view log messages or to change the severity filter.

Changes: Lists the changes from the last publish event.

Delivery Status: Lists the number of destinations updated.

Update History: Click “View all updates” for a detailed list.

Actions: Click the “Update Now” icon to update the web site.
Add website

This page allows you to add a Site Studio web site.

Site Details

User Name: The user name to access the Site Studio web site.

Password: The password to access the Site Studio web site.

The user name and password entered here defines what user account is used to access the Site Studio web site; and therefore controls whether you will be able to publish a site without getting access control errors.

Server CGI URL: The URL to the Content Server instance that is hosting your Site Studio web sites (e.g., http://server7/stellent/idcplg). Enter the Server CGI URL and click Connect. The Site ID drop-down list is automatically populated.

Site ID: The unique identifier of your Site Studio web site in the Content Server. Choose a Site ID from the drop-down list and click Generate Manifest URL. The Manifest URL field is automatically populated.

Site Name: Enter a descriptive name (e.g., “Business Website” or “ProductionServer7”).

Site Description: Enter any additional descriptive information here.

Delivery Options: Specify whether the Publishing Utility must check that all content has been delivered (required).

Select your delivery option:

- **Incremental**: Content will be delivered incrementally: that is, if delivery of a package of content items is interrupted, the content items successfully delivered and retrieved so far will not be discarded. Delivery of the remaining items will resume at the next delivery attempt, as dictated by the subscription's delivery rule. Use this option if delivering to an FTP server.

- **Atomic**: For each user, the Publishing Utility must successfully deliver all the items sent in each delivery; otherwise, no items will be delivered. (No incremental delivery.) Only available when delivering to a Subscription Client.

- **Synchronized**: The Publishing Utility must successfully deliver all the items sent in each delivery to all users; otherwise, no items will be delivered to any users. The Synchronized delivery option is useful when you need to reliably distribute identical copies of content, in parallel, to multiple servers (e.g., to an array of mirrored web servers that support load-balancing or failover.) Only available when delivering to a Subscription Client.

Preferred Page Extension: the default page extension is HTML. Change this if you want to uses a different page extension such as HCSP or JSP.

Update Schedule

When you specify your update schedule (start time, end time, and interval between deliveries), you can limit the time of day that content will be delivered. For example, if you specify delivery every 60 minutes, to begin at 8:00 AM and end at 5:00 PM, content will be delivered hourly—but not before 8:00 AM or after 5:00 PM.

**Every day**: Select this option to check the content every day of the year.

**Only on**: Select this option and then select the desired day(s) of the week to check content only on those days.

**Only on these dates**: Select this option to check content every month on certain dates. You can type days of the month as numbers separated by spaces. Use “last” to specify the last day of the
month. (For example, entering **10 20 last** sets checking for the 10th, 20th, and last day of the month.)

**Manual Update:** Select this option to prevent the site from being checked. As a result, no content will be automatically published. This option is useful if you want to manually check and publish the site (by clicking Update Now on the Status page).

**First Update at:** Specify the start time for checking to occur. Enter a time in hour and minutes and select **AM** or **PM** (for example, 3:40 PM).

**Update Every:** Specify the interval between checks. Type a two-digit number and select **Hours** or **Minutes** from the list (to check once a day, enter **24** and choose **Hours**).

**Last Update at:** Specify the final time for checking to occur. Enter a time as hour and minutes and select **AM** or **PM** (for example, 3:40 PM).

**Deliver To**

You can choose which destination to deliver to by checking the subscribe option for that destination, and unsubscribe by checking the unsubscribe option. Click **Save** when you are done defining your site details, update schedule, and delivery information.

---

## All destinations

This page allows you to perform these tasks:

- View the list of destinations
- Create a new destination by clicking **Create New Destination**.
- View the detail page of a destination by clicking its Name

---

## Create new destinations

This page allows you to update or delete current publish destination information or to add a new publish destination.

### Destination Details

**Destination Name:** Provide a descriptive name for this destination (required).

**Description:** You can add any descriptive information for this destination.

**Destination Type:** (required)

- **Subscription Client:** The content is delivered to a Subscription Client instance. You must specify Subscription Client Details below.
- **FTP Server:** The content is delivered directly to a designated FTP server. You must specify FTP Server Details below.

### Subscription Client Details

This section is for use when using a Subscription Client instance.

**Destination Push Url:** The URL of the hosting server where the Site Studio web site is to be published (e.g., http://productionserver7:8886 or http://106.106.106.1:8886).

**Password:** The password for the destination (Subscription Client). The password is used by the Subscription Client to authenticate itself to the Publishing Utility.
FTP Server Details
This section is for use when using an FTP server.

FTP Server Location: The URL of the FTP server (required).
Server Port Number: The FTP server port number.
FTP User Name: The FTP server login name (required).
FTP Password: The FTP server password.
FTP Subdirectory: A subdirectory to place content in.

Event logs

This page allows you to:
- View the list of event logs
- Start creating an event log by clicking Create Event Log
- View the detail page of an event log by clicking its Name

Primary Log
While the Publishing Utility is running, it automatically generates a log file named syndicator.log. This is an ASCII text file written to the Publishing Utility installation directory. In the Publishing Utility administration interface, this log file is referred to as the Primary log.

By default, the Publishing Utility uses a rotation method to manage its log file. Additional log files are created when the current one reaches its maximum configured size. The Publishing Utility continues to generate log files of a constant size until it has generated the number configured. When the specified number of files is reached, the Publishing Utility deletes the oldest log file when it creates the next new one. The rotating scheme provides control over exactly the amount of disk space used for storing log messages; and the most recent logging information is always retained.

The default file size for syndicator.log is 1024 kilobytes. The default number of files to rotate is 7; you can set this number as high as 100.

Database Log
This event log is required for the Current Status display. The default logging level is ‘Info’ to provide informational messages. The logging level can be changed to suit your needs. See “Create event log” on page 19 for more information.

Deleting Event Logs
To explicitly delete event logs:
1. On the Event Logs page, click the link for the log you want to delete.
2. On the resulting Event Log page, click Delete.

Note: You cannot delete the Primary log.

The Publishing Utility saves event and transmission messages generated by the Publishing Utility components, so that you can review them in case of problems. This section provides an overview of these logging features.
Create event log

This page allows you to create additional event logs.

**Log Details**

Enter a log name and select the desired logging levels.

**Log name:** Choose a name for this log. (You can choose the name arbitrarily, but it should normally reflect the log’s purpose.)

**Description:** Enter a description of this log’s purpose.

**Log Method:** Select a File or Custom log method.

- **File:** ASCII text file, with user-specified file name and rotation. You must specify File Logging details below.
- **Custom:** Custom log using a specified Java class API. You must specify Custom Logging details below.

**Logging Levels**

**Default Logging Level:** Sets the default error level for logging facilities.

You can set the level of events that are logged from various component facilities of the Publishing Utility. The severity levels are:

- **Critical:** Displays information of a serious error or crash.
- **Error:** Displays information when an operation has failed.
- **Warning:** Displays unusual conditions.
- **Info:** Displays informational messages.
- **Verbose:** Displays multiple-line status messages.
- **Debug:** Displays debug information for programmers. This logging level returns a large number of messages and can impact system performance. For this reason, only use debug to troubleshoot specific problems.

Use the facility error drop-down lists to override the default error logging level for particular logging facilities. See “Logging facilities” on page 20 for more information.

**Log Destinations**

For each log, the destination can be one of the following:

- **File Logging:** ASCII text file, with user-specified file name and rotation.
- **Database Logging:** Publishing Utility database, with user-specified purge cycle.
- **Mail Logging:** Email notification to a specified administrator address.
- **Custom Logging:** Custom log using a specified Java class API.

**File Logging**

Select the File Logging option if you want this log to send messages to a text file. This section of the page allows you to define the following settings:

**File name:** Enter a file name for this log.

**File handling:** Select one of the following options:

- **Overwrite log file:** Starts the log file fresh each time the Publishing Utility is started.
- **Append to existing log file:** Appends new entries to the existing log.
- **Rotate log files by size**: Causes the Publishing Utility to automatically generate additional log files when the current file reaches the threshold size specified. When the specified threshold is reached, the Publishing Utility will delete the oldest file as it creates the next new one.

**Rotate files every...KB**: Set the threshold size (in kilobytes) of the log file at which file rotation will occur. The maximum setting allowed is 1024 KB.

**Keep...old files**: Set the maximum number of files to retain in file rotation. The maximum setting allowed is 100.

Log files can become quite large. To generate smaller log files, use logging levels that generate messages less frequently, such as Critical or Error. Also, using the “Rotate log files by size” option for File handling ensures that the disk space used for logging remains constant.

**Custom Logging**

This log destination directs log messages to a Java routine for processing. For use with custom logging processors.

**Logging facilities**

These are the facilities for which event logging can be specified:

- **csm.sitestudio**: Content source monitor used to monitor Site Studio sites.
- **csm.web**: Content source monitor used to monitor web sites or FTP sites. When publishing a Site Studio site, many messages will show up as csm.web (see also csm.sitestudio).
- **database**: All of the server’s basic interactions with the database are logged here.
- **dataobject**: Messages related to the storing and retrieving of objects from the database.
- **date-time**: All messages associated with date or time conversions are logged here.
- **delivery**: Messages regarding the general aspects of content delivery, independent of delivery method.
- **delivery.ftp**: Messages specific to delivering content via FTP.
- **delivery.ice**: Messages specific to delivering content via ICE (note that both pull and push ICE subscriptions would log here).
- **event**: Notification of the creation, modification, and deletion of resources (such as offers, subscribers, and subscriptions), including the distribution of such events over the network.
- **httpd**: Connection and networking messages generated by the Publishing Utility’s built-in web server(s).
- **httpd.content**: Messages generated by the Publishing Utility’s built-in content server (for serving files from content source monitors).
- **httpd.tomcat**: Messages generated by Publishing Utility administration server.
- **ice**: The actual ICE messages exchanged between Publishing Utility and Subscription Client, as well as their processing.
- **ice-cache**: Messages concerning insertions and removals from the Publishing Utility’s ICE cache (a subsystem used to queue up items which will be sent to the Subscription Client, optimizing delivery whenever possible).
- **packagemanager**: Messages relating to the recording of content updates coming from source monitors.
replicator: Component used by csm.web to crawl web sites; also used by the Subscription Client. Triggers and content sinks will log in this facility. (Content sinks are modules for storing content in different repositories—the logical opposite of content source monitors.)

scheduler: Messages from the scheduler responsible for triggering timed updates, crawls, deliveries, and other periodic tasks.

security: General messages from the security and authorization subsystem.

soap.service: Describes SOAP (Simple Object Access Protocol) transaction activity.

syndicator: Messages produced by the general operation of the Publishing Utility itself (including most informational messages).

template: Messages produced by publishing utility’s sending of e-mail messages defined by event templates.

ui.validation: Messages relating to the interaction with a JavaBean.

ui.web: Web Interface log messages.

xml: Our XML/HTML parser; used in various places but most notably for handling ICE packages.

View log messages

This page allows you to view log messages. Show only log messages of severity higher than:

- **Critical:** Displays information of a serious error or crash.
- **Error:** Displays information when an operation has failed.
- **Warning:** Displays unusual conditions.
- **Info:** Displays informational messages.
- **Verbose:** Displays multiple-line status messages.
- **Debug:** Displays debug information for programmers. This logging level returns a large number of messages and can impact system performance. For this reason, only use debug to troubleshoot specific problems.

General settings

This page allows you to view and edit the general settings for your Publishing Utility.

Identity

These settings identify your Publishing Utility and determine its operational properties:

**Server Name:** The name of this Publishing Utility; usually your company’s name (required).

**Server UUID:** The Universal Unique Identifier (UUID) identifies your Publishing Utility. This entry is pre-populated. If this entry is blank, enter the license key number that is provided as part of your license agreement (required).

**Server Description:** Information about your site; usually a description of your business or other information. This description is written to the configuration file of the Subscription Client.

**Server URL:** The URL through which a user can log in to the Publishing Utility. This field is pre-populated with a generated URL depending on the configuration with which the Publishing
Utility was first started. However, this generated URL is just meant to be an example for the administrator. The administrator should verify the URL and modify it as needed.

**Server URL for Subscription Client [ICE]:** The Information and Content Exchange (ICE) URL through which a user can log into this Publishing Utility to download content. This field is pre-populated with a generated URL, depending on the configuration with which Publishing Utility was first started. However, this generated URL is just meant to be an example for the administrator. The administrator should verify the URL and modify it as needed.

**Administration**

Your administrator password can be changed by entering a new password and clicking **Save**.

**Database Purge**

These settings determine when and how often the Publishing Utility purges inactive content from its database. Purging the database is a two-phase process. The first phase is the marking of inactive content items, which are items that have seen no activity in the number of hours specified. The second phase deletes database information and occurs on the days and times specified.

The Publishing Utility does not delete the actual content files, only stored information about them. After a purge, elements that have not changed—that you want to appear consistently in content, such as a graphic of your corporate logo—must be added to the database again.

Purge time settings determine the window for running purges and the time interval between selecting for stale offers. By default, the Publishing Utility runs purges every 5 minutes for 24 hours (from 12:00 A.M to 11:59 P.M.). When entering times, use a 12-hour clock with either AM or PM.

**Purge content items older than:** The hours specifies the number of hours of inactivity after which the content items are marked for deletion.

If you leave this field empty, purges never occur. If you fill it in, purges will occur on the days and times that you specify below—deleting items older than the number of hours that you specify in this field.

**Purge logs and package update history entries older than:** The hours specifies the number of hours of inactivity after which the logs and package update history are marked for deletion.

If you leave this field empty, purges never occur. If you fill it in, purges will occur on the days and times that you specify below—deleting items older than the number of hours that you specify in this field.

**Purge Frequency:** These settings determine the frequency of purges. They can be set to occur daily, on specific days of the week, or on specific dates of the month.

- **Every day:** The Publishing Utility runs the purge process every day of the year.
- **Only on:** The Publishing Utility purges eligible content on the days of the week selected.
- **Only on these dates:** The Publishing Utility purges eligible content on the dates entered in the field. Type dates as numbers separated by spaces. Use “last” to specify the last day of the month. (For example, entering **10 20 last** sets checking for the 10th, 20th, and the last day of the month.)
- **Manual Update:** The Publishing Utility will not automatically purge the database. This option is useful if you want to manually purge the database.

**First purge at:** Sets the earliest time of day that the purge process can run.

**Purge every:** Sets the interval between purges, expressed in minutes or hours. Enter a two-digit number in the field and select Hours or Minutes. To purge once a day, enter 24 hours. By default, the Publishing Utility purges every thirty (30) minutes.
**Last purge at:** Sets the latest time of day that the purge process can run

**Time Format**

The Time Format section allows you to determine the display format of all time settings used in the Publishing Utility. You must type all times, including those in a 24-hour/12-hour format, in the format you choose. The Time Format section provides these options:

- The 24-hour clock option is used in conjunction with Local or GMT settings and causes times to be displayed in 24-hour format.
- The Local time option displays times in the local system time. This is the default setting.
- The GMT (Greenwich Mean Time) option displays times in Greenwich Mean Time.

The GMT +/- options allow you to display times offset from GMT by the number of hours and minutes entered in the text boxes.

**Note:** You must type times into the Publishing Utility in the same format in which you have chosen to display them.
CHAPTER 4

Configuration (XML) elements

The Publishing Utility provides an XML-based language for filtering content. This section provides a reference guide to the replication configuration XML schema used by the Publishing Utility. Each XML element is documented, followed by the DTD documenting the XML schema. This is intended to assist those responsible for programming and configuring extensions for software.

In this section:
- Base elements
- Filters
- Triggers
- Sinks
- DTD for job and filter elements

Base elements

This section describes the configuration elements that are used to implement transformations, filters, triggers, and sinks.

defaults element

The defaults element specifies the elements to apply to a job if they are not explicitly set within the job elements themselves. The defaults child elements are ice-delivery-rule (1), filterset (0 to many), trigger (0 to many), and sink (0 to many).

job element

The job element causes a job to execute at the time specified by the job’s delivery rule. The job element requires a defined URL where the content resides. The following attributes allow specification of the information needed to complete a job.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>lurl</td>
<td>(required) URL of the Publishing Utility ICE server.</td>
<td>[string]</td>
</tr>
<tr>
<td>username</td>
<td>User ID for password-protected sites.</td>
<td>optional</td>
</tr>
<tr>
<td>password</td>
<td>Base-64 encoded password for password-protected sites.</td>
<td>optional</td>
</tr>
<tr>
<td>baseref</td>
<td>Specifies the virtual path where the site is hosted. For example, to replicate the web site <a href="http://www.stellent.com">http://www.stellent.com</a> and have it re-served by the local web server as <a href="http://target.domain/cool/Stellent/index.html">http://target.domain/cool/Stellent/index.html</a>, baseref would be defined as /cool/Stellent, and C:\MySrv\Web\docroot\cool\Stellent as the localdir (see below).</td>
<td>optional</td>
</tr>
</tbody>
</table>
### Site Studio Publishing Utility - Administration Guide

**Configuration (XML) elements**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
<th>Type</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>user-agent</td>
<td>Identification of the web browser (user agent) that the Publishing Utility “mimics” when crawling the site.</td>
<td>[string]</td>
<td>Default: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)</td>
</tr>
<tr>
<td>max-depth</td>
<td>Maximum number of levels (depth) of links to follow when retrieving content from a site. For example, if max-depth is set to 1 and the target URL is <a href="http://www.stellent.com">http://www.stellent.com</a>, links found on the page <a href="http://www.stellent.com">http://www.stellent.com</a> index.htm are followed to the next level down, all target files are downloaded, and the process ends. With max-depth set to 2, secondary links are followed to the third level down, all target files are downloaded, and the process ends. A value of -1 is interpreted as no limit.</td>
<td>[integer]</td>
<td>Default: -1</td>
</tr>
<tr>
<td>max-pages</td>
<td>Maximum number of pages to be downloaded. A value of -1 is interpreted as no limit.</td>
<td>[integer]</td>
<td>Default: -1</td>
</tr>
<tr>
<td>absolute-to-relative</td>
<td>true converts fully qualified and absolute URLs to URLs relative to the top level of the content. This allows content to be browsed without error after download. false leaves URLs intact.</td>
<td>[string]</td>
<td>Default: true</td>
</tr>
<tr>
<td>pnmtohttp</td>
<td>HTTP URL that points to the root directory of the remote web site’s RealAudio server. The replication engine translates PNM (Real Audio) URLs to HTTP URLs using this setting. This translation enables the replication engine to retrieve the RealAudio file(s) from the specified PNM URL and to calculate the appropriate local file path.</td>
<td>[string]</td>
<td>Example: pnm:\IP_address where IP_address is the IP address of your Real-Audio server.</td>
</tr>
<tr>
<td>pnmtopnm</td>
<td>PNM (RealAudio) URL that points to the local Real-Audio server. The replication engine uses this setting to transform downloaded RealAudio files so they can be served from the local RealAudio server.</td>
<td>[string]</td>
<td>Example: pnm:\IP_address where IP_address is the IP address of your Real-Audio server.</td>
</tr>
<tr>
<td>mmstohttp</td>
<td>HTTP URL that points to the root directory of the remote web site’s Microsoft NetShow server. The replication engine translates MMS (NetShow) URLs to HTTP URLs using this setting. This translation enables the replication engine to retrieve the RealAudio file from the PNM URL and to calculate the appropriate local file path.</td>
<td>[string]</td>
<td>Example: pnm:\IP_address where IP_address is the IP address of your NetShow server.</td>
</tr>
<tr>
<td>mmstomms</td>
<td>MMS (NetShow) URL that points to the local Microsoft NetShow server. The replication engine uses this setting to transform downloaded NetShow files so they can be served from the local NetShow server.</td>
<td>[string]</td>
<td>Example: pnm:\IP_address where IP_address is the IP address of your NetShow server.</td>
</tr>
<tr>
<td>subscription-id</td>
<td>ID of the subscription; defined by the Publishing Utility.</td>
<td>[string]</td>
<td></td>
</tr>
<tr>
<td>subscription-state</td>
<td>As a job executes, its state changes. This element specifies the latest job state within an offer.</td>
<td>[string]</td>
<td></td>
</tr>
</tbody>
</table>
The child elements of the job element are localdir(1), ice-delivery-rule (1), trigger (0 to many), filterset (0 to many), and sink (0 to many). Only localdir is required.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>offer-name</td>
<td>Name of the offer.</td>
</tr>
</tbody>
</table>
| atomic-use| **true**: Forces the replication engine to download all the content for a given delivery. If there any error is encountered while downloading, the download directory will be reverted to its previous state.  
**false**: The replication engine will download all the content it can, even if it encounters errors while downloading other content.  
Is overridden by enable-rollback if the enable-rollback attribute is defined. Default: false |
| editable  | **true**: Indicates that the subscriber may edit/alter the content before using it. **false** (or unspecified): The subscriber is expected to use the content without any alteration. Default: false |
| enable-rollback | **true**: Forces the replication engine to download all the content in a transaction (as with the atomic-use attribute).  
**false**: The replication engine will download content normally (as an incremental download).  
This value overrides the value of the atomic-use attribute. If enable-rollback is defined, its value takes precedence. Default: false |
| ip-status | One of: **PUBLIC-DOMAIN**: content has no licensing restrictions.  
**FREE-WITH-ACK**: only licensing restriction is requirement to display an acknowledgement of the content source.  
**SEE-LICENSE**: content has licensing restrictions as already agreed to in an existing licensing agreement.  
**SEVERE-RESTRICTIONS**: this flag should not be used routinely. It is a red flag for an administrator on the subscriber site.  
**CONFIDENTIAL**: content is confidential and must be specially protected. Default: [string] |
| rights-holder | Original source of the distribution rights. Default: [string] |
| showcredit | **true**: Indicates that the subscriber is explicitly expected to acknowledge the source of the data. **false**: Subscriber does not need to acknowledge the source of the data. Default: false |
| usage-required | **true**: Subscriber is expected to return usage data regarding ultimate viewers of the distributed content. **false**: Subscriber need not return usage data regarding ultimate viewers of the distributed content. Default: false |
localdir child element
The localdir element is used to specify the directory on the local system to which content will be downloaded. Typically, this is the root document directory of the local web server that will re-serve the site. The localdir element has no child elements and just one attribute.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>dir</td>
<td>(required) path to the local directory</td>
<td>[pathname]</td>
</tr>
</tbody>
</table>

ice-delivery-rule child element
This element specifies the schedule for running this job. If no attributes are set for ice-delivery-rule, the job is run immediately (in the order in which it appears in the configuration file), in pull delivery mode every five minutes (the default); otherwise, the job is run according to the schedule described by ice-delivery-rule.

An ice-delivery-rule defines the year(s), month(s), date(s) and day(s) of the week on which the job runs, starting and ending times for the job, and the frequency at which the job runs. If the job does not have an ice-delivery-rule child element, the ice-delivery-rule specified in the defaults element is used.

The ice-delivery-rule for subscriptions in siclone.config, without a subscription-id of “PROVIDER,” is dictated by Publishing Utility. If any changes are made, they are overwritten at the next status check.

The following explanation of ice-delivery-rule attributes is based on section 4.2.2 of the Information and Content Exchange (ICE) Protocol’s W3C Submission available from http://www.w3.org/TR/NOTE-ice.

All of these attributes are conceptually joined by AND (as opposed to OR). That is, within a single ice-delivery-rule, valid delivery times satisfy all the conditions listed in the attributes within the rule.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>mode</td>
<td>specifies the delivery mode for the subscription: pull: the replication engine polls the Publishing Utility for new or updated content and retrieves it push: the Publishing Utility determines when content has changed and sends the package to the replication engine when appropriate</td>
<td>Default: pull</td>
</tr>
<tr>
<td>duration</td>
<td>delivery cannot take place after this amount of time past the starttime.</td>
<td>Has no meaning if starttime has not been specified. See “ICE Duration format” below.</td>
</tr>
<tr>
<td>maxfreq</td>
<td>maximum amount of time between content updates.</td>
<td>See “ICE Duration format” below.</td>
</tr>
<tr>
<td>minfreq</td>
<td>minimum amount of time between content updates.</td>
<td>See “ICE Duration format” below.</td>
</tr>
</tbody>
</table>
ICE DateTime format

When a date and time need to be specified in a single string, the ICE DateTime format is:
CCYY-MM-DDThh:mm:ss.s

Where C, Y, M, D, and T are upper case letters.

ICE Duration format

When a period of time needs to be specified, the ICE Duration format is: PnS (where P and S are upper case letters and n is the number of seconds).

Filters

A filter element describes a transformation that is applied to a downloaded content file. A filterset contains filter elements, and each filter is applied in the order the filter elements appear.
in the filterset. Several types of filter can be specified. Specification of a filter element depends on the kind of filter used, as specified by the type attribute.

These elements are described:

- filterset element (child of job element)
- filter element (child of filterset element)
- authentication element (child of filterset element)

**filterset element (child of job element)**

This element is used to specify the attributes against which the source content is compared. If there are no filterset elements, the default filtersets (defined by the defaults element) are used. For more information, see “defaults element” on page 25.

Each attribute of a filterset is compared to the relevant characteristics of the source content. Each and every attribute must match for the filterset itself to be considered a match. The filtersets are evaluated in the order in which they appear in siclone.config, and the first filterset match is applied. If a filterset has child elements, they are searched if the parent filterset has tested true, allowing the user to specify mutually exclusive filtersets by placing more specific filtersets inside more general ones.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>The replication engine uses filtersets in different contexts: when applying filters to a downloaded file, when deciding whether a given URL should be downloaded, or when attempting to authenticate a password-protected URL. The type attribute specifies the context in which the filterset should be invoked</td>
<td>[filter or exclude or include or authenticate or transform-link]</td>
</tr>
<tr>
<td></td>
<td>filter — for URLs that match this filterset, the child filter elements are applied during download, transforming the content in the manner specified by these filters.</td>
<td>Default: filter</td>
</tr>
<tr>
<td></td>
<td>exclude — for URLs that match this filterset, content is not downloaded.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>include — for URLs that match this filterset, content is downloaded, even if it would normally be excluded.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>authenticate — for URLs that match this filterset, use the child authenticate element for authentication instead of the default username and password associated with the parent job element.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>transform-link — you can rewrite links on content being downloading by creating a filterset element with type=&quot;transform-link&quot;. Any link whose URL matches the filterset will be rewritten depending on the filterset attributes defined below.</td>
<td></td>
</tr>
</tbody>
</table>
**action**
- **added** — matches URLs that are being downloaded for the first time.
- **modified** — matches URLs that already exist.
- **removed** — matches URLs that are no longer used. Filtersets that contain this action are applied after the job has finished running. This attribute is only used with the filter type of filterset.

**hostname**
A match if the URL’s host name (the domain name part of the URL) matches the specified host name

**port**
Considered a match if the URL’s port number matches the specified port number

**path**
A wildcard pattern to match the file path of the URL (the part following the URL’s hostname) currently being downloaded

**pathfiltertype**
Specifies whether the path attribute is a file wildcard or a Perl regular expression pattern. If the path specifies a wildcard pattern, Java regular expression syntax is used for pattern matching by default. Use the following reference:

http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html

**mimetype**
A match if the URL’s MIME-type starts with the value specified by this type; therefore, text matches both text/html and text/xml

**job-attribute-name**
Specifies which attribute name to match on; should be used in conjunction with **job-attribute-value**

**job-attribute-value**
Specifies the value of the attribute to match on; must be used in conjunction with **job-attribute-name** (see above)

By specifying **job-attribute-name** and **job-attribute-value** the filterset will only apply to those jobs that have the specified attribute name value pair. For example:

```xml
<filterset job-attribute-name="offer-name" job-attribute-value="sports news">...
</filterset>
```

will only apply to jobs that have the attribute offer-name="sports news"

**[Any other attribute]**
For any other attribute present in the element, the filterset will compare the value of this attribute with the value of item’s metadata attribute with the same name. If item has an attribute with that name, and its value does not match, the item will not pass the filter; otherwise, the rest of the filterset will be evaluated.
Some examples:

```xml
<filterset type="transform-link" path="*.subdir/" mimetype="" transform-hook="com.kinecta.examples.TransformhookExample" transform-hook-param="optional param used by the TransformhookExample constructor" />

<filterset type="transform-link" path="*.subdir/" mimetype="" convert-absolute-to-relative="true" />

<filterset type="transform-link" path="*.subdir/" mimetype="" replace-url="/page-not-available.html" />
```

### filter element (child of filterset element)

A filter element describes a transformation that is applied to a downloaded file. A filterset contains filter elements and each filter is applied to a downloaded file in the order the filter elements appear in the filterset. Several filters can be specified.

Global attributes, listed below, can be applied to any filter element, regardless of the type. The rest of the content of a filter element depends on the kind of filter used, as specified by the type attribute.

Three attributes (change-path-regex, change-extension, and append-extension) are used to rename the output of a filter or filters. If several filters are run, the settings contained in the last filter take effect and the output filename is determined by these values.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>(required) classifies the type of filter</td>
<td>regex or cmd or java or xsl</td>
</tr>
</tbody>
</table>
The following filter types are supported:

**regex filter type**

Regular expression substitutions can be applied to a document using PERLs substitution syntax. The content of the filter element is treated as a PERL substitution. (See [http://www.perl.org/](http://www.perl.org/) for more information on the syntax of per substitutions.)

Example (UNIX-specific):
```
<filter type="regex">
  s/img/object/sg
  <!-- replace every instance of "img" with "object" -->
</filter>
```

**cmd filter type**

The cmd filter is invoked from the command line; it invokes a separate process. The downloaded file is used as input. The input and output are specified by passing in the corresponding file pathname on the command line. The Publishing Utility does this by searching for the string **%%fi** in the command line and substituting it with the path to a temporary input file containing the content and substituting the string **%%fo** with the path to an output file whose contents are sent to the next filter. It then executes the command line.

There are additional variables that you can use within the definition of the cmd filter: **%%ff** will be transformed into the name of the file being downloaded, **%%o** will be translated into the name of the offer containing this item, **%%u** will be the URL of the original item, and **%%aXXXXX** will find the attribute named XXXXX in the job element and substitute the value of that attribute.
Example (note that this command line is operating system dependent):
<filter type="cmd" use-input-as-output="false">
 /cmd/customfilter.pl %%fi %%fo
 <!--invoke a custom Perl script -->
</filter>

Note: When executing cmd filters that use DOS shell commands, you must invoke the DOS shell by using the following syntax:
cmd /C

Following this, you may use any DOS command known to the shell.

The following attributes apply only to cmd filters:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>error-exit-code</td>
<td>specifies the command return code that represents failure (error)</td>
<td>optional</td>
</tr>
<tr>
<td>success-exit-code</td>
<td>specifies the command return code that represents success (no error)</td>
<td>optional</td>
</tr>
<tr>
<td>use-input-as-output</td>
<td>true: the input file is used as the output file; false: the input file remains intact and the output file is given a different name</td>
<td>optional [boolean]</td>
</tr>
<tr>
<td></td>
<td>Default: true</td>
<td></td>
</tr>
</tbody>
</table>

java filter type
You can specify a class file that implements the com.kinecta.replicator.ContentFilter interface. The class file must be accessible through the Java classpath. Each time this filter is invoked on a downloaded file, it instantiates an object of the class and invokes the filterContent() method on it, allowing you to create arbitrarily complex filters.

The content of the filter element is treated as the name of the Java class to run. This must be a fully qualified class name; that is, it must include the name of the package (if any).

Example:
<filter type="java">
 adc.extras.TextOnlyFilter
 <!--invoke a custom java filter -->
</filter>

xsl filter type
You can apply an XSL style sheet to any XML or HTML file being downloaded by creating a filter with and setting the type to “xsl”. The XML or HTML file is replaced by the output from the XSL processor.

Example:
<filter type="xsl" src="stylesheet.xsl">
</filter>

The following attribute applies only to XSL filters:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>src</td>
<td>(required) location of the XSL stylesheet.</td>
<td>valid local file pathname or URL.</td>
</tr>
</tbody>
</table>
authentication element (child of filterset element)

The authentication element specifies the user name and password for a password-protected URL. It has no child elements and requires two attributes.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>(required) user ID (name)</td>
<td>string</td>
</tr>
<tr>
<td>password</td>
<td>(required) user password</td>
<td>string</td>
</tr>
</tbody>
</table>

Triggers

Triggers enable the replication engine to run commands either before a package is downloaded or after the download is complete. Triggers are defined in the Subscription Client configuration file.

A trigger is defined by a trigger element, which is contained within the context of a job element.

There are three types of trigger:

- cmd triggers
- http-post triggers
- Java triggers

You can define other attributes specific to the trigger, or embed content within the trigger element, as necessary.

cmd triggers

A cmd trigger allows a shell script or an operating system command to be run as a trigger. The command is passed to the operating system. The syntax is:

```
<trigger when="before|after"
  type="cmd"
  [ignore-failure="true|false"]
  [name="triggername"]
  run-on-abort="true|false"
  run-if-unchanged="true|false">
  [command line arguments to be passed to the operating system]
</trigger>
```

- **when:** (required) specifies when the trigger should run—before a package is downloaded, or after.
- **type:** (required) specifies the type of trigger as a cmd (command line) trigger.
- **ignore-failure:** specifies whether Subscriber requires successful execution of the trigger (see “Java triggers” on page 37 for more information):
  - **true:** considers the package download successful even if the trigger fails.
  - **false:** (default) considers the package download unsuccessful if the trigger fails.
- **name:** specifies a name for the trigger.
- **run-on-abort:** specifies whether or not the command should be run, even if the job encountered errors. By default, this is false.
run-if-unchanged: specifies whether or not the trigger should run even if there has been no change in content. By default, this is false.

The following example of a DOS command-line trigger renames any file with the filename extension of .xml to use the filename extension .html:
<trigger when="after" type="cmd">
    cmd /C rename *.xml *.html
</trigger>

To run this trigger in the directory where content files have been downloaded, you must use %\%d, followed by `\` in a Windows environment or `/` in a UNIX environment, to represent the name of the download directory.

The following DOS-specific example changes the filename extensions of all .xml files in the download directory to the HTML filename extension:

```bash
cmd /C ren %\%d\*.xml *.html
```

There are additional variables that you can use within the definition of the cmd trigger:
- %\%o — will be translated into the name of the offer containing this item.
- %\%aXXXXX — will find the attribute named XXXXX in the job element, and substitute the value of that attribute.

http-post triggers

The http-post trigger sends any file to an URL location using HTTP POST functionality. The syntax is:
<trigger when="before|after"
    type="http-post"
    url="URL"
    file="filename"
    [ignore-failure="true|false"]
    [name="triggername"] />

when: (required) specifies when the trigger should run—before a package is downloaded or after

type: (required) specifies the type of trigger as an http-post

url: specifies the URL to which you post the file named in the file attribute

file: the full path of the file to be downloaded

ignore-failure: an optional entry that specifies whether the replication engine requires successful execution of the trigger (see “Java triggers” on page 37 for more information):
- true: considers the package download successful even if the trigger fails.
- false: (default) considers the package download unsuccessful if the trigger fails.

name: specifies an optional name for the trigger

One possible use of http-post is to send a copy of the log file to another location.

For example:
<trigger when="after" type="http-post"
    url="http://support.stellent.com/rcv_log"
    file="c:\Program Files\Stellent\Stellent Subscription Client\subscriber.log">
Java triggers

Java triggers are java classes that are invoked by defining a configuration entry as follows:

```
<trigger when="before|after"
  type="java"
  code="classpath"
  [ignore-failure="true|false"]
  [name="triggername"] />
```

**when:** (required) specifies when the trigger should be fired: before a package is downloaded, or after

**type:** (required) specifies the type of trigger as a Java class

**code:** (required) fully specified class name of the Java class to loaded. It must implement the TriggerProcess interface: (see the section “Programming Java triggers” on page 37)

**ignore-failure:** specifies whether the replication engine requires successful execution of the trigger.

- **true:** considers the package download successful even if the trigger fails
- **false:** (default) considers the package download unsuccessful if the trigger fails*

**name:** specifies a name for the trigger

If a trigger fails, the replication engine default action is to stop processing triggers. If the Publishing Utility has requested confirmation, the replication engine sends an ICE message (430) indicating that package transmission has failed. Because ICE state has not been updated, the package is requested again. These actions do not take place if the ignore-failure entry is set to true.

Programming Java triggers

To implement a Java trigger, a Java class need only implement the TriggerProcess interface. This interface defines one method:

```
public boolean trigger (SiteReplicator repl, org.w3c.com.Element triggerTag, boolean aborted, boolean changeHappened)
```

**SiteReplicator repl:** the SiteReplicator that invoked this trigger; can be used to retrieve the context in which the trigger was invoked

**org.w3c.com.Element triggerTag:** the trigger element that specified this trigger; can be used to retrieve other attributes specified in the element

**boolean aborted:** true if the package delivery was aborted; always set to false for triggers called before delivery starts.

**boolean changeHappened:** true if the subscription has changed since the last delivery; is always set to false for triggers run before replication starts

Return true if the trigger was processed successfully or return false on failure. If the trigger returns false and is associated with a subscription that requires confirmation, a confirmation failure (ICE code 430) message is sent.

Defined triggers are run at each package download. However, if the Publishing Utility breaks a full package into several smaller packages, the triggers are applied to each of these smaller packages.
Sample

The following sample is from SimpleTriggerProcess.java:

```java
/** This class is a simple trigger demonstration **/  
package com.kinecta.util;
import com.kinecta.replicator.*;
import java.awt.*;
import org.w3c.dom.Element;
public class SimpleTriggerProcess implements TriggerProcess
{
    // must support a no-argument constructor, as this gets loaded
    // via Class.newInstance()
    public SimpleTriggerProcess()
    {}  
    /** Implementation of trigger */
    public boolean trigger(SiteReplicator repl, Element triggerTag, 
    boolean aborted, boolean changeHappened)
    {
        Toolkit.getDefaultToolkit().beep();
        System.out.println("THIS SPACE FOR RENT");
        return true;
    }
}
```

Sinks

Sink elements represent ContentSink objects which can be used to store content in the repository of your choice. If no sink is specified, the replication engine will place the downloaded content onto the file system. The syntax is:

```xml
<sink factory="xxx.xxx.xxx.Factory"/>
```

Other sink-specific attributes or content can be added to the element and read by the content sink. The factory attribute must contain the fully qualified class path of a Java class that implements the com.kinecta.replicator.ContentSinkFactory interface.

Multiple sinks can be specified for a job element, allowing the content to be stored in multiple locations. If you wish to download content to a custom content sink and the file system, you must include a reference to the default content sink provided with the replication engine:

```xml
<sink factory="my.custom.ContentSinkFactory"/>
<sink factory="com.kinecta.util.FileSystemSink"/>
```

The default content sink for the replication engine takes an extra attribute, download-directory. If this attribute is specified, content will be downloaded to the specified directory. If it is not specified, the content sink will use the localdir element for that job element. If you wish to download to multiple locations on the file system, specify one or more FileSystemSinks, each with a different download-directory attribute.
DTD for job and filter elements

This configuration file fragment contains the XML Document Type Definition (DTD) for the job and filter elements.

```xml
<!ELEMENT defaults (ice-delivery-rule?, filterset*, trigger*, sink*)>
<!ELEMENT job (localdir+, ice-delivery-rule?, trigger*, filterset*, sink*)>
<!ELEMENT localdir EMPTY>
```

-- ice-delivery-rule, filtersets that appear here are default for jobs -->

-- ice-delivery-rule, filtersets that appear here are local to this job (override the default) -->

```
<!ATTLIST job
  %BASIC;
  url %URI; #REQUIRED
  username CDATA #IMPLIED
  password CDATA #IMPLIED

  <!-- http website crawling attributes -->
  <!ATTLIST job
  baseref CDATA #IMPLIED
  dust %BOOLEAN_ENUM; "no"
  user-agent CDATA "Mozilla/4.x (Win95)"
  max-depth CDATA "-1"
  max-pages CDATA "-1"
  absolute-to-relative %BOOLEAN_ENUM; "true"
>

  <!-- multimedia attributes -->
  <!ATTLIST job
  pnmtohttp CDATA #IMPLIED
  pnmtopnm CDATA #IMPLIED
  mmstohttp CDATA #IMPLIED
  mmstomms CDATA #IMPLIED
>

  <!-- attributes specific to ICE subscription jobs -->
  <!ATTLIST job
  subscription-id  CDATA  #IMPLIED
  subscription-state CDATA  #IMPLIED
  offer-name  CDATA  #IMPLIED
  atomic-use  (false | true) "false"
  editable  (false | true) "false"
  ip-status CDATA  #IMPLIED
  rights-holder CDATA  #IMPLIED
  showcredit  (false | true) "false"
  usage-required  (false | true) "false"
>

<!ELEMENT localdir EMPTY>
```

-- from the ice spec: see http://www.w3.org/TR/NOTE-ice -->

```
<!ELEMENT ice-delivery-rule EMPTY>
```

-- from the ice spec: see http://www.w3.org/TR/NOTE-ice -->
<!ELEMENT filterset (filterset* | filter* | authentication?)>
<!ATTLIST filterset %BASIC;
type (filter | exclude | include | authentication) "filter"
action NMTOKENS "added modified"
hostname CDATA #IMPLIED
port CDATA "-1"
path CDATA "+*"
pathfiltertype (wildcard | perl) "wildcard"
mimetype CDATA "text"
>
<!ELEMENT authentication EMPTY>
<!ATTLIST authentication %BASIC;
username CDATA #REQUIRED
password CDATA #REQUIRED
>
<!ELEMENT filter (#PCDATA | replace+)>
<!ATTLIST filter %BASIC;
type (java | perl | cmd | XSL) #REQUIRED
src %URI; #IMPLIED
>
<!-- attributes for filter when type="cmd" -->
<!ATTLIST filter success-exit-code CDATA #IMPLIED
error-exit-code CDATA #IMPLIED
use-input-as-output %BOOLEAN_ENUM; "true"
>
<!ELEMENT sink EMPTY>
<!ATTLIST sink %BASIC;
factory CDATA "com.kinecta.util.FileSystemSink"
download-directory CDATA #IMPLIED
>
APPENDIX A

Configuration file properties

The entries in the startup configuration file can be edited to control the operational characteristics of the Publishing Utility. For most implementations, this file will not need to be edited and many of the properties will not be applicable to your server environment. The configuration file is located in the Publishing Utility installation directory.

In this section:
- The root element
- The options element
- The services element
- The database element
- The content-sources element
- The extensions element
- The ldap element
- The j2ee element

The root element

The root element of the configuration file must always be syndicator. These elements can appear as child elements:
- options
- services
- database
- content-sources
- extensions.

The options element

This section covers the options element and the following sub-elements:
- The timeFormat element
- The log element
- The proxy element
- The ssl element

Attributes of the options element

The options element contains global settings that govern how the Connection Server behaves. This element contains the following attributes:
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>browser-path</td>
<td>Specifies the path to the web browser you use to administer Connection Server. This is required only if you wish to automatically launch your browser when the Connection Server starts and if the Connection Server is unable to find it.</td>
<td>Valid file path. Default: The path to the default browser defined to the operating system.</td>
</tr>
<tr>
<td>content-browsing</td>
<td>Specifies whether a web browser connecting to the Content Server can view files listed in directories</td>
<td>true: Browsing enabled false: Browsing disabled Default: true</td>
</tr>
<tr>
<td>custom-item-fields</td>
<td>A comma-delimited list of custom metadata properties that, if present, you would want to send alongside the content item in an ICE package. If the metadata property is not listed, or if this attribute is not present, the metadata will not be sent—despite the fact that it was associated with a content item by its content source.</td>
<td>Your list can contain a maximum of 26 properties.</td>
</tr>
<tr>
<td>default-pull-delivery-rule-name</td>
<td>This attribute define the names of the default pull delivery rule. The default delivery rules have special limitations and behaviors: for example, they can not be deleted or renamed. The pull/push attributes allow you to change which named delivery rules have these special conditions.</td>
<td>Default: Default Delivery Rule</td>
</tr>
<tr>
<td>default-push-delivery-rule-name</td>
<td>This attribute define the names of the default push delivery rule. The default delivery rules have special limitations and behaviors: for example, they can not be deleted or renamed. The pull/push attributes allow you to change which named delivery rules have these special conditions.</td>
<td>Defaults: Default Push Delivery Rule</td>
</tr>
<tr>
<td>download-base</td>
<td>Specifies the local directory under which all locally served content is stored</td>
<td>Valid file path</td>
</tr>
<tr>
<td>Attribute</td>
<td>Purpose</td>
<td>Values and Default</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| filewatcher-checksum  | Specifies the calculation of checksums on files to determine if a file has changed; replaces the use of the timestamp to determine changes to files. If very large files are being distributed, the checksum calculation may cause a negative performance impact. | **true**: Checksum calculation enabled  
**false**: Checksum calculation disabled.  
Default: false                                                                                                                                 |
| hostname              | Allows you to override the default hostname that is automatically determined at startup.                                                                                                                                 | **Valid hostname.**  
**Default**: Determined at startup  
To find the current hostname, check this line at Connection Server start-up:  
“Starting Connection Server on <hostname>” |
| ice-cache-silence-time-out | If this attribute is enabled (and caching is enabled), the Connection Server considers that an error has occurred if the cache does not detect any new item added to, or removed from, the database within the specified time. The Connection Server then refuses any connection and returns an ICE code 503 error message until new activity is detected. It also logs this error in the log file. | **0**: disabled  
**>0**: specified time in milliseconds  
Default: 0                                                                                                                                 |
| ice-cache-size        | Specifies the maximum number of ICE items that the Connection Server’s cache can contain                                                                                                                                                                       | **Default**: 5000                                                                 |
| ice-cache-update-interval | Specifies in milliseconds how often the Connection Server queries the database to update its ICE items cache                                                                                                                                               | **Default**: 10000 (10 seconds)                                                   |
| ice-failover-adjustment | If this attribute is enabled when a Subscription Client connects to an ICE server running as a backup server, or if a Subscription Client was previously connected to an ICE server running as a backup server, the Connection Server rolls back the subscription state by the number of milliseconds specified. As a result, redundant adds or removes may be sent. | **0**: disabled  
**>0**: number of milliseconds to roll back  
Default: 0                                                                                     |
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>intra-distributed-name</td>
<td>Represents the name of this machine within a distributed-architecture environment. This name is for use solely among the Connection Server installations on the machines.</td>
<td>Default: There is no default value.</td>
</tr>
<tr>
<td>MAC-address</td>
<td>In order to generate UUIDs for users, Connection Server needs a MAC address. Under most circumstances, Connection Server will automatically detect your MAC address. However, there may be instances where the Connection Server cannot detect it. In those situations, you can use this attribute to manually insert it.</td>
<td>Format: 6 pairs of hexadecimal digits, separated by colons: XX:XX:XX:XX:XX Each X represents a hexadecimal digit 0-9 or A-F Example: 9f:6a:03:4f:0d:09</td>
</tr>
<tr>
<td>max-package-size</td>
<td>Specifies the maximum number of items that can be delivered in any given ICE package.</td>
<td>Default: 500</td>
</tr>
<tr>
<td>max-push-bps</td>
<td>Dictates the maximum rate at which the Connection Server will push ICE packages to subscribers. Value is in bytes per second. A value of -1 means no limit.</td>
<td></td>
</tr>
<tr>
<td>max-push-package-creation-threads</td>
<td>Specifies the maximum number of threads to use for generating packages to distribute via push.</td>
<td>Default: No limit is imposed.</td>
</tr>
<tr>
<td>max-push-retry</td>
<td>Specifies the maximum number of milliseconds that Connection Server waits before retrying a connection to Subscription Client when attempting a push mode delivery of content. The Connection Server, by default, makes several attempts at short time intervals prior to using the value in this configuration setting.</td>
<td>Default: 300000 (5 minutes)</td>
</tr>
<tr>
<td>max-push-threads</td>
<td>Maximum number of threads a push mode delivery opens.</td>
<td>Default: 6</td>
</tr>
<tr>
<td>min-push-threads</td>
<td>Minimum number of threads a push mode delivery keeps open.</td>
<td>Default: 3</td>
</tr>
<tr>
<td>push-connect-timeout</td>
<td>Specifies the amount of time in milliseconds allowed for push to connect to a client before timing out.</td>
<td>Default: 30000</td>
</tr>
</tbody>
</table>
This example of an options element sets two options:

<options browser-path="/usr/local/bin/netscape" content-browsing="false">

### The timeFormat element

This attribute specifies parameters for the display and entry of time data. These settings are configurable on the Connection Server General Settings interface page.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>use24hour</td>
<td>Specifies whether to use a 24-hour clock format or a 12-hour format</td>
<td><strong>true</strong>: Use 24-hour format</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>false</strong>: Use 12-hour format</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: false</td>
</tr>
<tr>
<td>time_format</td>
<td>Specifies the time to use, either localTime or GMT</td>
<td>Default: localTime</td>
</tr>
<tr>
<td>sign</td>
<td>Time can be specified as an offset of hours and minutes from GMT with a sign + or -</td>
<td>+ adds hours/minutes to the GMT time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- subtracts hours/minutes from the GMT time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: blank</td>
</tr>
<tr>
<td>hours</td>
<td>Specifies the number of hours to add/subtract from the GMT time for the offset</td>
<td>Default: blank</td>
</tr>
<tr>
<td>minutes</td>
<td>Specifies the number of minutes to add/subtract from the GMT time for the offset</td>
<td>Default: blank</td>
</tr>
</tbody>
</table>
The log element

The table below shows the attributes for the log sub-element. The log element has no child elements.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Specifies the minimum priority of log messages written to the log. See “Logging levels” on page 46 for additional information on message priorities.</td>
<td>Default: info</td>
</tr>
<tr>
<td>overwrite</td>
<td>Selects whether new logs are appended to the existing log. Otherwise, the log file is overwritten each time Connection Server starts.</td>
<td>true: Overwrites the existing log file each time Connection Server starts false: Appends log messages to the existing log file. Default: false</td>
</tr>
</tbody>
</table>

Logging levels

You can set the level of events that are logged from various component facilities of Connection Server. The severity level values are:

- **debug**: for programmers
- **verbose**: multiple-line status messages
- **info**: informational messages
- **warning**: unusual conditions
- **error**: an operation has failed
- **critical**: serious error or crash

This example of a log child element sets the default logging level for all facilities and the logging level for two facilities:

```
<options>
  <log default="warning" ice="critical" replicator="debug"/>
</options>
```

This example sets the default logging level to **warning**, suppressing messages of lower severity and returning messages of higher severity. Messages from the ICE facility are suppressed unless they are of **critical** priority, while all events, including debugging information, are logged for the replicator facility.

In general, we recommend that you set logging levels for facilities by means of the Connection Server user interface, rather than by adding tags to the configuration file (as in the example above). If you do set these levels through the configuration file, your entries there override any logging levels that you set through the user interface. You should generally add `log default` tags to the configuration file only if you are trying to capture additional information about an error while starting the server.
## Logging facilities

The following are the facilities for which event logging can be specified:

<table>
<thead>
<tr>
<th>Facility</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>analyzer</td>
<td>Component used by the webcrawler (csm.web) that analyzes web pages, JavaScript, etc., when crawling a site for download.</td>
</tr>
<tr>
<td>audit</td>
<td>Logs authorization events, and provides an audit trail of who changed what content.</td>
</tr>
<tr>
<td>auto-upgrade</td>
<td>Subscriber component that handles software updates delivered over the Internet.</td>
</tr>
<tr>
<td>csm.dirs</td>
<td>Content source monitor (csm) used to watch directories of files (also referred to as filewatcher).</td>
</tr>
<tr>
<td>csm.files</td>
<td>Content source monitor used to monitor sets of individual files (occasionally referred to as item level offers).</td>
</tr>
<tr>
<td>csm.web</td>
<td>Content source monitor used to monitor websites or ftp sites.</td>
</tr>
<tr>
<td>database</td>
<td>All of the server's basic interactions with the database are logged here (except for things like custom database content source monitors).</td>
</tr>
<tr>
<td>dataobject</td>
<td>Messages concerning the various entities (offers, subscriptions, packages) contained in the system and their interaction with the database.</td>
</tr>
<tr>
<td>date-time</td>
<td>All messages associated with date or time conversions are logged here.</td>
</tr>
<tr>
<td>delivery</td>
<td>Messages regarding the general aspects of content delivery, independent of delivery method.</td>
</tr>
<tr>
<td>delivery.ftp</td>
<td>Messages specific to delivering content via FTP.</td>
</tr>
<tr>
<td>delivery.ice</td>
<td>Messages specific to delivering content via ICE (note that both pull and push ICE subscriptions would log here).</td>
</tr>
<tr>
<td>delivery.mail</td>
<td>Messages specific to delivering content via e-mail.</td>
</tr>
<tr>
<td>ejb</td>
<td>Messages relating to the Connection Server’s interaction with a J2EE application server.</td>
</tr>
<tr>
<td>event</td>
<td>Notification of the creation, modification, and deletion of resources (such as offers, subscribers, and subscriptions)—including the distribution of such events over the network.</td>
</tr>
<tr>
<td>filter</td>
<td>Messages generated by content filtering.</td>
</tr>
<tr>
<td>httpd</td>
<td>Connection and networking messages generated by Connection Server’s built-in web server(s).</td>
</tr>
<tr>
<td>httpd.content</td>
<td>Messages generated by Connection Server’s built-in Content Server (for serving files from content source monitors).</td>
</tr>
<tr>
<td>httpd.tomcat</td>
<td>Messages generated by Connection Server’s administration server.</td>
</tr>
<tr>
<td>ice</td>
<td>The actual ICE messages exchanged between Connection Server and Subscription Client, as well as their processing.</td>
</tr>
</tbody>
</table>
Facility | Description
--- | ---
**ice-cache** | Messages concerning insertions and removals from the Connection Server’s ICE cache (a subsystem used to queue up items which will be sent to the Subscription Client, optimizing delivery whenever possible).

**KTL** | Messages generated by the Kinecta Transformation Language.

**ldap** | Messages generated by the LDAP synchronization subsystem while communicating with an LDAP server.

**logging** | Messages generated by the logging system itself, for logging to files such as databases and e-mail.

**login** | Logs user attempts to log on to the Connection Server.

**replicator** | Component used by csm.web to crawl websites; also used by the Subscription Client. Triggers and content sinks will log in this facility. Content sinks are modules for storing content in different repositories—the logical opposite of content source monitors.

**scheduler** | Records times of content updates, delivery, etc.

**security** | General messages from the security and authorization subsystem.

**serializer** | Messages generated by component that handles transaction boundaries for content source monitors.

**soap** | Describes SOAP (Simple Object Access Protocol) transaction activity.

**sweeper** | Cleans up after the Replicator.

**syndicator** | Messages produced by the general operation of the Connection Server itself (including most informational messages).

**template** | Messages produced by Connection Server’s sending of e-mail messages defined by event templates.

**ui** | Messages from the administration JSP pages themselves (not the TomCat server).

**xml** | Our xml/html parser, used in various places but most notably for handling ice packages.

---

**The proxy element**

The table below shows the attributes for the proxy sub-element. The proxy element has no child elements.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>host</strong></td>
<td>Specifies the IP address or host name of the proxy server</td>
<td>Valid IP address or host name for the proxy server Default: null (-1)</td>
</tr>
<tr>
<td><strong>password</strong></td>
<td>Specifies the password for proxy authentication.</td>
<td>Valid password for secure proxy server.</td>
</tr>
</tbody>
</table>
Some HTTP proxy servers require a username and password. The Connection Server can use these proxy servers for ICE push delivery if you include the `username` and `password` tags in the configuration file’s `proxy` element. The Connection Server supports HTTP proxy authentication for ICE push only. An example proxy element would look like this:

```xml
<options>
  <proxy host="proxy.stellent.com" port="3128" proxyset="true"
        username="user" password="password"/>
</options>
```

**Note:** The web crawler cannot do proxy authentication

## The `ssl` element

The table below shows the attributes for the `ssl` (Secure Socket Layer) sub-element. The `ssl` element has no child elements.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>port</code></td>
<td>Specifies the port number of the proxy server</td>
<td>Valid port number. Default: null (-1)</td>
</tr>
</tbody>
</table>
| `proxyset` | Specifies whether proxy server settings are set: | true: Proxy server settings are set  
false: Proxy server settings are not set  
Default: false |
| `ssl-port` | Specifies the port number used by the proxy server for Secure Sockets Layer (SSL) requests. | Valid port number. Default: null (-1) |
| `username` | Specifies the user name for proxy authentication. | Valid user ID for secure proxy server. |

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
</table>
| `enable`  | Specifies whether SSL security will be used by the Publishing Utility. Set this attribute to "true" to enable SSL. | true: The Publishing Utility will use SSL security to connect to the Administration user interface and the ICE server.  
false: The Publishing Utility can only use non-SSL ports. Default: false |
| `required` | Specifies whether only SSL security can be used and prohibits the use of other protocols. | true: Only SSL can be used. Non-SSL connection are not supported.  
false: Both SSL and non-SSL connections are supported. Default: false |
Site Studio Publishing Utility - Administration Guide

Site Studio Publishing Utility no longer provides Secure Sockets Layer (SSL) certificates. You must create a SSL certificate and edit the configuration file to enable SSL. An example ssl element would look like this:

```xml
<options>
<ssl enable="true" required="false" admin="true"
keystorefilename="yourfilename" keystorepass="yourpassword"/>
</options>
```

The services element

This section covers the services element and following sub-elements:

- The admin-server attribute
- The content-server element
- The admin-server attribute
- The ice-server element
- The master-server element

In this example for configuring services, the ICE Server is auto-started and the other two servers are not. Port numbers are assigned:

```xml
<services>
<ice-server port="8890" />  
<content-server start="false" port="8891"/>  
<admin-server start="true" port="8889" />  
<master-server receiver1="http://slave1:8890"/>
</services>
```

The admin-server attribute

This attribute specifies operational parameters for the Administration Server, which allows Connection Server to be administered through by the browser-based Administrator interface.
### Site Studio Publishing Utility - Administration Guide

#### Note:

Once **max-thread-queue** is exceeded, the service responds to new requests with the message: 500 Server Too Busy

---

**The content-server element**

Specifies operational parameters for the standard HTTP server that serves the content files.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>master-url</strong></td>
<td>Specifies the URL of the master host from which this Content Server instance receives event notification. If null or not specified, the service does not attempt to connect to a master.</td>
<td>URL:port</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Port default is 8880</td>
</tr>
<tr>
<td><strong>max-threads</strong></td>
<td>Specifies the maximum number of request-handling threads this service can create; the number of threads specified is never exceeded.</td>
<td>Integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 30</td>
</tr>
<tr>
<td><strong>max-thread-queue</strong></td>
<td>Specifies the maximum allowable number of backlogged requests.</td>
<td>Integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 400</td>
</tr>
<tr>
<td><strong>min-threads</strong></td>
<td>Specifies the initial number of request-handling threads this service will create when started; the number of threads specified always start.</td>
<td>Integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 2</td>
</tr>
</tbody>
</table>

---
The ice-server element

Specifies operational parameters for the ICE Server—the service that uses the ICE protocol to handle offers, subscriptions, and updates.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>master-url</td>
<td>Specifies the URL of the master host from which this ICE Server receives event notification. If null or not specified, the service does not attempt to connect to a master.</td>
<td>URL:port Port default is 8880</td>
</tr>
<tr>
<td>max-threads</td>
<td>Specifies the maximum number of request-handling threads this service can create; the number of threads specified is never exceeded.</td>
<td>Integer Default: 7</td>
</tr>
</tbody>
</table>
The master-server element

Specifies operational parameters for event distribution—the service that allows notifications to be sent between Connection Server components in multiple-host environments.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>max-thread-queue</td>
<td>Specifies the maximum allowable number of backlogged requests.</td>
<td>Integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 30</td>
</tr>
<tr>
<td>min-threads</td>
<td>Specifies the initial number of request-handling threads this service will create when started; the number of threads specified always start.</td>
<td>integer</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 7</td>
</tr>
<tr>
<td>port</td>
<td>Specifies the port on which the ICE Server service should listen.</td>
<td>A valid port number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 8890</td>
</tr>
<tr>
<td>self-url</td>
<td>Specifies the URL and port this host uses to receive event notifications from the master host. Recommended for use when the IP address may not reliably point to the same machine, such as in some DHCP environments.</td>
<td>A fully-qualified URL and port number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example: <a href="http://your_url:7575">http://your_url:7575</a></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If null or not specified, the master determines a value, based on the IP address.</td>
</tr>
<tr>
<td>ssl-port</td>
<td>Specifies the port on which the ICE Server listens for communications protected by Secure Sockets Layer (SSL) security.</td>
<td>A valid port number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 8893</td>
</tr>
<tr>
<td>start</td>
<td>Specifies that the ICE Server service should start automatically.</td>
<td>true: ICE Server starts automatically</td>
</tr>
<tr>
<td></td>
<td></td>
<td>false: ICE Server does not automatically start</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: true</td>
</tr>
<tr>
<td>accept-unlisted</td>
<td>Specifies whether or not the master (event sending) host should allow unlisted slave (receiving) hosts to register for event notification. Unregistered slaves are not automatically reconnected when the master restarts and, therefore, are not automatically synchronized.</td>
<td>true: Unregistered slaves are accepted for connection to a master</td>
</tr>
<tr>
<td></td>
<td></td>
<td>false: Only registered slaves are accepted for connection to a master</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: false</td>
</tr>
<tr>
<td>authentication-id</td>
<td>This attribute allows you to specify a login that the various servers in the distributed architecture will use to authenticate with each other.</td>
<td>Connection Server logins/ usernames should contain only ASCII characters</td>
</tr>
</tbody>
</table>
The database element

Most JDBC driver documentation details the Java code needed for a program to connect to a specific JDBC driver. In general, there are four common interactions. Each of these interactions can be specified in the configuration file; not all interactions are needed for each driver.

- Selecting the JDBC driver to be used
- Choosing how the driver is to be registered
- Selecting the JDBC URL used to connect to the database
- Setting additional parameters not specified in the JDBC URL

The database element and its child elements, driver, user, and property, enable Connection Server to connect to a database using a JDBC driver.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication-pass</td>
<td>This attribute allows you to specify a password that the various servers in the distributed network will use to authenticate with each other.</td>
<td>Connection Server passwords should contain only ASCII characters.</td>
</tr>
<tr>
<td>master-url</td>
<td>Allows two master servers to communicate with each other, which might be necessary if changes are potentially happening on multiple machines.</td>
<td>A fully-qualified URL and port number.</td>
</tr>
<tr>
<td>port</td>
<td>Specifies the port on which the master host listens.</td>
<td>A valid port number Default: 8880</td>
</tr>
<tr>
<td>receivern</td>
<td>Specifies a list of URLs for event-receiving slave hosts. If accept-unlisted is false, each slave must be listed to receive events. This is the recommended configuration.</td>
<td>receivern, where ( n ) is an integer value starting with 1. Default: none</td>
</tr>
<tr>
<td>self-url</td>
<td>Allows two master servers to communicate with each other, which might be necessary if changes are potentially happening on multiple machines.</td>
<td>A fully-qualified URL and port number.</td>
</tr>
<tr>
<td>shutdown-slaves</td>
<td>Specifies whether or not the slaves should be notified when the master server shuts down or restarts. If true, the various slaves will also either be shutdown or restarted.</td>
<td>Default: true</td>
</tr>
<tr>
<td>start</td>
<td>Specifies that the master service should start automatically.</td>
<td>true: Master service starts automatically false: Master service does not automatically start Default: true</td>
</tr>
</tbody>
</table>
The description of these elements is shown in the following example. The database section of the configuration file appears in the configuration file provided with the Connection Server software.

```xml
<database type="dbtype" max-connections=9/>
<driver jdbcURL="jdbc:oracle:kinecta_ice"
driver="oracle.jdbc.driver.OracleDriver"/>
<user username="user" password="pass"/>
<property key="debug-level" value="2"/>
<property key="keyname" value="5"/>
</database>
```

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>max-connections</td>
<td>The maximum number of database connections allowed.</td>
<td>whole number &gt;0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default: 9</td>
</tr>
<tr>
<td>max-retries</td>
<td>The number of times a database operation will be retried after one fails.</td>
<td>Default: 3</td>
</tr>
<tr>
<td>retry-delay</td>
<td>The amount of time in milliseconds between database retries.</td>
<td>amount of time in milliseconds between database retries</td>
</tr>
</tbody>
</table>

The content-sources element

The content-source element allows the addition of a custom content source monitor. The factory element is a child of this element. For more information and an example, refer to the Replication Customization Guide.
The extensions element

This section covers the extensions element and following sub-element:

- The factory element

The extensions element is used to add custom extensions to the Connection Server. Its inclusion is wholly optional and most often will enable the addition of custom content source monitors and adapters. The SDK contains examples of such. The extensions element specifies a class that implements `com.kinecta.syndicator.extension`. This element has a child element: `factory` (any number, including none).

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>class</td>
<td>Java class name to be invoked for an extension.</td>
<td>valid Java class name for this extension</td>
</tr>
<tr>
<td>param</td>
<td>Information to be passed to the Java class when invoked.</td>
<td>any string you wish to pass to the extension</td>
</tr>
</tbody>
</table>
The factory element

The factory element implements com.kinecta.syndicator.extensionfactory and can be a child of both the extensions element and the content-sources element. It has no child elements.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>class</td>
<td>Java class name to be invoked for a factory.</td>
<td>valid Java class name for this factory</td>
</tr>
<tr>
<td>param</td>
<td>Information to be passed to the Java class when invoked.</td>
<td>any string you wish to pass to the factory</td>
</tr>
</tbody>
</table>

The ldap element

The ldap element causes the Connection Server to launch in LDAP mode. This mode allows the Connection Server to import and delete user records by validation against an LDAP (Lightweight Directory Access Protocol) database.

Note: The Connection Server does not directly authenticate against the LDAP database

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldap-enabled</td>
<td>Tells the Connection Server whether to run in LDAP mode or not.</td>
<td>true or false (no default)</td>
</tr>
<tr>
<td>ldapURL</td>
<td>Specifies the LDAP server’s IP address and port.</td>
<td>ldap://your_IP_address:389</td>
</tr>
<tr>
<td>username</td>
<td>Specifies an LDAP administrator user name, in order to establish a JNDI connection.</td>
<td>The user name with administrator permissions. For the iPlanet LDAP server, default is: “cn=Directory Manager”</td>
</tr>
<tr>
<td>userpassword</td>
<td>Specifies an LDAP administrator password, in order to establish a JNDI connection.</td>
<td>The password corresponding to the user name above.</td>
</tr>
<tr>
<td>rolebase</td>
<td>Specifies the LDAP element that forms the base of the search for matching roles.</td>
<td>Default: “cn=KinectaAdminGroup, ou=Groups, dc=ldap_server_domain_name_element,... dc=ldap_server_domain_name_element”</td>
</tr>
<tr>
<td>rolename</td>
<td>Specifies the name of the LDAP server attribute that contains the role name.</td>
<td>Example: “cn”</td>
</tr>
<tr>
<td>rolesearch</td>
<td>Specifies the LDAP search pattern for selecting roles in the LDAP realm.</td>
<td>Example: “cn” rolesearch=“(uniquemember= {0})”</td>
</tr>
<tr>
<td>digest</td>
<td>Specifies the digest algorithm used to store passwords.</td>
<td>Default: “CLEAR”</td>
</tr>
</tbody>
</table>
The j2ee element

This section covers the j2ee element and following sub-elements:
- The knet element
- The license-server host element
- The knet-server host element
- The soap-server host element
- The tracking-server host element
- The clickthru-server element

Attributes of the j2ee element

The j2ee element governs how the Connection Server behaves when run as a web application within the (J2EE-compliant) WebLogic application server. This element contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>rolesubtree</td>
<td>Specifies whether to search subelements.</td>
<td>true if you want role searches to search the subtrees of elements selected by rolebase; false if you want to search only the top-level elements.</td>
</tr>
<tr>
<td>passwordname</td>
<td>Specifies the name of the LDAP server attribute that contains the password.</td>
<td>Form depends on the digest attribute’s setting. Default: passwordname=&quot;userpassword&quot;</td>
</tr>
<tr>
<td>userpattern</td>
<td>Specifies the search pattern for selecting users in the LDAP realm.</td>
<td>Use {0} as a shorthand for the distinguished name (dn) pattern corresponding to the users you want to retrieve. Example: userpattern=&quot;uid={0},ou=People,dc=your_domain,dc=com&quot;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Purpose</td>
<td>Values and Default</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td>initContextFactory</td>
<td>The name of the class that implements the naming context to facilitate the lookup and discovery of objects using JNDI services. This class name will vary depending on the JNDI services used.</td>
<td></td>
</tr>
<tr>
<td>appServerURL</td>
<td>The complete URL (including the protocol and port) to access the application server. This URL is used to look up EJBs and other resources from the application server.</td>
<td>Default: t3://localhost:7001/</td>
</tr>
<tr>
<td>docRoot</td>
<td>The document root of the web application. The web application serves JSP and HTML pages to the client from this location.</td>
<td>Example: installation_path/webpages/syndicator (where installation_path specifies the local directory in which the Connection Server is installed)</td>
</tr>
<tr>
<td>resourcePath</td>
<td>The root of the installed web application. The web application uses this path to look up other resources (such as configuration files).</td>
<td>Example: installation_path</td>
</tr>
<tr>
<td>ejbAccessUser</td>
<td>Together with ejbAccessUserPasswd, specifies the user credentials required to look up the EJBs in the application server. If you change the System Administrator login/password in the web application, you should change these defaults to match.</td>
<td>Default: administrator</td>
</tr>
<tr>
<td>ejbAccessUserPasswd</td>
<td>Password corresponding to the value of ejbAccessUser.</td>
<td>Default: administrator</td>
</tr>
<tr>
<td>virtualDir</td>
<td>The name of the web application. Used for administrative purposes. Should match the name by which the web application is deployed in the application server.</td>
<td>Example: virtualDir=ConnectionServer</td>
</tr>
</tbody>
</table>
The knet element

The knet element directs the Connection Server to an appropriate Sellent Subscription Client Deployment Server. The knet element has five child elements: license-server host, knet-server host, soap-server host, tracking-server host, and clickthru-server.

The license-server host element

The license-server host child element specifies host information for the service that checks for licenses when the Connection Server and Subscription Client start up.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies URL to the license-server host.</td>
<td>http://your_url</td>
</tr>
<tr>
<td>port</td>
<td>Specifies port used to access the license server.</td>
<td>Configurable, but usually shares the same port as the application server.</td>
</tr>
</tbody>
</table>

The knet-server host element

The knet-server host child element specifies host information for the Subscription Client Deployment Server itself.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies the URL to the Subscription Client Deployment Server host.</td>
<td>http://your_url</td>
</tr>
<tr>
<td>port</td>
<td>Specifies port used to access the Subscription Client Deployment Server.</td>
<td>Configurable, but usually shares the same port as the application server.</td>
</tr>
</tbody>
</table>

The soap-server host element

The soap-server host child element specifies host information for the SOAP (simple Object Access Protocol) server. The SOAP protocol is used for communication between the Subscription Client Deployment Server and the Connection Server.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies URL to the SOAP server’s host.</td>
<td>http://your_url</td>
</tr>
<tr>
<td>port</td>
<td>Specifies port used to access the SOAP server.</td>
<td>Configurable, but usually shares the same port as the application server.</td>
</tr>
</tbody>
</table>
The tracking-server host element

The tracking-server host child element identifies the server that hosts the Content Matrix service.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies URL to the Content Metrics server’s host.</td>
<td>http://your_url</td>
</tr>
<tr>
<td>port</td>
<td>Specifies port used to access the Content Metrics server.</td>
<td>Configurable, but usually shares the same port as the application server.</td>
</tr>
</tbody>
</table>

The clickthru-server element

The clickthru-server child element identifies the server that hosts the Content Metrics click-through tracking service.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Purpose</th>
<th>Values and Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies URL to the Content Metrics click-through server’s host.</td>
<td>http://your_url</td>
</tr>
<tr>
<td>port</td>
<td>Specifies port used to access the Content Metrics click-through server’s host.</td>
<td>Configurable, but usually shares the same port as the application server.</td>
</tr>
</tbody>
</table>
THIRD PARTY LICENSES

OVERVIEW

This appendix includes a description of the Third Party Licenses for all the third party products included with this product.

- Apache Software License (page A-1)
- W3C® Software Notice and License (page A-2)
- Zlib License (page A-4)
- General BSD License (page A-5)
- General MIT License (page A-5)
- Unicode License (page A-6)
- Miscellaneous Attributions (page A-7)

APACHE SOFTWARE LICENSE

* Copyright 1999-2004 The Apache Software Foundation.

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

* http://www.apache.org/licenses/LICENSE-2.0

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

**W3C® Software Notice and License**

* Copyright © 1994-2000 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University).
* All Rights Reserved. http://www.w3.org/Consortium/Legal/

* This W3C work (including software, documents, or other related items) is
* being provided by the copyright holders under the following license. By
* obtaining, using and/or copying this work, you (the licensee) agree that
* you have read, understood, and will comply with the following terms and
* conditions:

* Permission to use, copy, modify, and distribute this software and its
* documentation, with or without modification, for any purpose and without
* fee or royalty is hereby granted, provided that you include the following
* on ALL copies of the software and documentation or portions thereof,
* including modifications, that you make:

* 1. The full text of this NOTICE in a location viewable to users of the
   redistributed or derivative work.

* 2. Any pre-existing intellectual property disclaimers, notices, or terms
and conditions. If none exist, a short notice of the following form
(hypertext is preferred, text is permitted) should be used within the
body of any redistributed or derivative code: "Copyright ©
Institute of Technology, Institut National de Recherche en
Informatique et en Automatique, Keio University). All Rights
Reserved. http://www.w3.org/Consortium/Legal/

3. Notice of any changes or modifications to the W3C files, including the
date changes were made. (We recommend you provide URIs to the location
from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT
NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE
ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR
DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising
or publicity pertaining to the software without specific, written prior
permission. Title to copyright in this software and any associated
documentation will at all times remain with copyright holders.
ZLIB LICENSE

* zlib.h -- interface of the 'zlib' general purpose compression library

version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org
Mark Adler madler@alumni.caltech.edu
GENERAL BSD LICENSE

Copyright (c) 1998, Regents of the University of California
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

"Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

"Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

"Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GENERAL MIT LICENSE

Copyright (c) 1998, Regents of the Massachusetts Institute of Technology

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

UNICODE LICENSE

UNICODE, INC. LICENSE AGREEMENT — DATA FILES AND SOFTWARE


NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright © 1991-2006 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.
THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

________________________
Unicode and the Unicode logo are trademarks of Unicode, Inc., and may be registered in some jurisdictions. All other trademarks and registered trademarks mentioned herein are the property of their respective owners.

MISCELLANEOUS ATTRIBUTIONS

Adobe, Acrobat, and the Acrobat Logo are registered trademarks of Adobe Systems Incorporated.

FAST Instream is a trademark of Fast Search and Transfer ASA.

HP-UX is a registered trademark of Hewlett-Packard Company.

IBM, Informix, and DB2 are registered trademarks of IBM Corporation.

Jaws PDF Library is a registered trademark of Global Graphics Software Ltd.

Kofax is a registered trademark, and Ascent and Ascent Capture are trademarks of Kofax Image Products.

Linux is a registered trademark of Linus Torvalds.

Mac is a registered trademark, and Safari is a trademark of Apple Computer, Inc.

Microsoft, Windows, and Internet Explorer are registered trademarks of Microsoft Corporation.

MrSID is property of LizardTech, Inc. It is protected by U.S. Patent No. 5,710,835. Foreign Patents Pending.

Oracle is a registered trademark of Oracle Corporation.

Portions Copyright © 1994-1997 LEAD Technologies, Inc. All rights reserved.

Portions Copyright © 1990-1998 Handmade Software, Inc. All rights reserved.

Portions Copyright © 1988, 1997 Aladdin Enterprises. All rights reserved.
Third Party Licenses

Portions Copyright © 1997 Soft Horizons. All rights reserved.
Portions Copyright © 1995-1999 LizardTech, Inc. All rights reserved.
Red Hat is a registered trademark of Red Hat, Inc.
Sun is a registered trademark, and Sun ONE, Solaris, iPlanet and Java are trademarks of Sun Microsystems, Inc.
Sybase is a registered trademark of Sybase, Inc.
UNIX is a registered trademark of The Open Group.
Verity is a registered trademark of Autonomy Corporation plc
Index

A
absolute-to-relative job element 26
accept-unlisted 53
action attribute 31
admin 50
admin-server attribute 50
    max-thread-queue 51
    max-threads 51
    min-threads 51
    port 51
    start 51
analyzr 47
append to existing log file 19
append-extension attribute 33
appServerURL 59
atomic delivery option 16
atomic-use job element 27
Attribute
    admin-server 50
attribute, admin-server 45, 50
audit 47
authentication
    described 35
authentication element 35
    password 35
    username 35
authentication-id 53
authentication-pass 54
auto-upgrade 47

B
base elements
    defaults 25
    described 25
    job 25
baseref job element 25
browser-path 42

C
change-extension attribute 33
change-path-regex attribute 33
class 56
    factory element
        class 57
clickthr-thru-server element 61
cmd filter
    error-exit-code 34
    success-exit-code 34
    use-input-as-output 34
cmd filter type 33
cmd trigger 35
    syntax 35
configuration file
    admin-server attribute 45, 50
content server element 51
database element 54
DTD for filter element 39
DTD for job element 39
ice-server 52
log element 46
options element 41, 60
proxy element 48
root element 41
ssl element 49
content-browsing 42
content-server element 51
    master-url 51
    max-thread-queue 51
    max-threads 51
    min-threads 51
    port 52
    remote-host 52
    self-url 52
    ssl-port 52
    start 52
content-sources element 55
critical 19, 46
csm.dirs 47
csm.files 47
csm.sitestudio logging facility 20
csm.web 47
csm.web logging facility 20
custom logging 19, 20

database 47
database element 54
driver 55
jdbcURL 55
max-collection 55
max-retries 55
password 55
retry-relay 55
username 55
database logging 19
database logging facility 20
database purge 22
dataobject 47
dataobject logging facility 20
date-time 47
date-time format 29
date-time logging facility 20
debug 19, 46
default 46
defaults base element 25
delivery 47
delivery logging facility 20
delivery options
  atomic 16
  incremental 16
  synchronized 16
delivery rules, ICE 28
delivery.ftp 47
delivery.ftp logging facility 20
delivery.ice 47
delivery.ice logging facility 20
delivery.mail 47
destination type
  FTP server 17
  Subscription Client 17
digest 57
dir attribute 28
docRoot 59
documentation 7
download-base 42
driver 55
DTD
  filter element 39
  job element 39
duration format 29

E
editable job element 27
ejb 47
ejbAccessUser 59
ejbAccessUserPasswd 59
Element
  clickthru-server 61
  content-server 51
  content-sources 55
  database 54
  factory 57
  ice-server 52
  j2ee 58
  knet 60
  knet-server host 60
  ldap 57
  license-server host 60
  log 46
  master-server 53
  options 41
  proxy 48
  root 41
  services 50
  soap-server host 60
  ssl 49
  timeFormat 45
  tracking-server host 61
elements
  content server 51
  database 54
  ice-server 52
  log 46
  options 41, 60
  proxy 48
  root 41
  ssl 49
enabled 49
error 19, 46
error-exit-code attribute 34
event 47
event logging facility 20
extension element
  class 56
  param 56
F
facilities
  logging levels 46
facility
  analyzer 47
  audit 47
  auto-upgrade 47
csm.dirs 47
csm.files 47
csm.web 47
database 47
dataobjects 47
date-time 47
delivery 47
delivery.ftp 47
delivery.ice 47
delivery.mail 47
ejb 47
event 47
filter 47
httpd 47
httpd.content 47
httpd.tomcat 47
ICE 20, 47
ice-cache 48
KTL 48
ldap 48
logging 48
login 48
replicator 48
scheduler 48
security 48
serializer 48
soap 48
sweeper 48
syndicator 48
template 48
ui 48
xml 48
factory element 57
file handling
  overwrite log file 19
  rotate log files by size 20
  file logging 19
  filewatcher-checksum 43
filter 47
  authentication child element 35
  filter element
    append-extension 33
    change-extension 33
    change-path-regex 33
    described 32
    no-save 33
    src 33
    type 32
  filter type
    cmd 33
    java 34
    regex 33
    xsl 34
filters
  configuring 29
  described 29
filterset element 30
  action 31
  described 29
  hostname 31
  path 31
  pathfiltertype 31
  port 31
FTP server
  location 18
  password 18
  port number 18
  subdirectory 18
  user name 18
FTP sever 17

H
host 48
hostname 43
hostname attribute 31
httpd 47
httpd logging facility 20
httpd.content 47
httpd.content logging facility 20
httpd.tomcat 47
httpd.tomcat logging facility 20
http-post trigger 36
syntax 36

I

ICE 20, 47
date-time format 29
duration format 29
icache 48
ice-cache logging facility 20
ice-cache-silence-time-out 43
ice-cache-size 43
ice-cache-update-interval 43
ice-delivery-rule
  maxfreq 28
  minfreq 28
  monthday 29
  startdate 29
  starttime 29
  stopdate 29
  weekday 29
ice-delivery-rule element 28
ice-failover-adjustment 43
ice-server element 52
  master-url 52, 54
  max-thread-queue 53
  max-threads 52
  min-threads 53
  port 53
  self-url 53, 54
  ssl-port 53
  start 53
incremental delivery option 16
info 19, 46
initContextFactory 59
ip-status job element 27

J

j2ee element 58
  appServerURL 59
  docRoot 59
  ejbAccessUser 59
  ejbAccessUserPasswd 59
  initContextFactory 59
  resourcePath 59
virtualDir 59
java filter type, described 34
java trigger 37
  programming 37
  syntax 37
JDBC 54
JDBC, about 54
jdbcTemplate 55
job base element 25
job element 26
  absolute-to-relative 26
  atomic-use 27
  baseref 25
  editable 27
  filterset child element 30
  ice-delivery-rule child element 28
  ip-status 27
  localdir child element 28
  max-depth 26
  max-pages 26
  mmstohttp 26
  offer-name 27
  password 25
  pnmtopnm 26
  pnmtopnm 26
  rights-holder 27
  showcredit 27
  subscription-id 26
  subscription-state 26
  url 25
  usage-required 27
  user-agent 26
  username 25
job schedule 15
job state 15

K

keystorefilename 50
keystorepass 50
knet element 60
knet-server host element 60
KTL 48
L
last update 15
ldap 48
ldap element 57
digest 57
ldap-enabled 57
ldapURL 57
passwordname 58
rolebase 57
rolename 57
rolesearch 57
rolesubtree 58
username 57
userpassword 57
userpattern 58
ldap-enabled 57
ldapURL 57
license-server host element 60
localdir child element 28
dir attribute 28
log destinations
append to existing log file 19
custom logging 19
database logging 19
file logging 19
mail logging 19
log element 46
default 46
overwrite 46
logging 48
Logging Facilities 47
logging facilities
csm.sitestudio 20
csm.web 20
database 20
dataobjects 20
date-time 20
delivery 20
delivery.ftp 20
delivery.ice 20
event 20
httpd 20
httpd.content 20
httpd.tomcat 20

ice-cache 20
packagemanager 20
replicator 21
scheduler 21
security 21
soap.service 21
syndicator 21
template 21
ui.validation 21
ui.web 21
xml 21
Logging Levels 46
logging levels
critical 19, 46
debug 19, 46
error 19, 46
info 19, 46
verbose 19, 46
warning 19, 46
logging levels, by facility 46
login 48

M
mail logging 19
manual update 17
master-server element 53
accept-unlisted 53
authentication-id 53
authentication-pass 54
receivern 54
shutdown-slaves 54
start 54
master-url 51, 52, 54
max-connections 55
max-depth job element 26
maxfreq attribute 28
max-package-size 44
max-pages job element 26
max-push-retry 44
max-push-threads 44
max-retries 55
max-thread-queue 51, 53
max-thrads 51, 53
minfreq attribute 28
min-push-threads 44
min-threads 51, 53
mmstohttp job element 26
mmstomms 26
mmstomms job element 26
monthday attribute 29

N
no-save attribute 33

O
offer-name job element 27
options element 41, 60
browser-path 42
content-browsing 42
download-base 42
filewatcher-checksum 43
hostname 43
ice-cache-silence-time-out 43
ice-cache-size 43
ice-cache-update-interval 43
ice-failover-adjustment 43
max-package-size 44
max-push-retry 44
max-push-threads 44
min-push-threads 44
run-interactive 45
start-browser 45
overrite log file 19
overwrite 46

P
packagemanager logging facility 20
param 56
factory element
param 57
password 48, 55
password job element 25
passwordname 58
path attribute 31
pathfiltertype attribute 31
pnmtohttp job element 26
pnmtopnm job element 26
port 49, 51, 52, 53
port attribute 31
proxy element 48
host 48
password 48
port 49
proxyset 49
ssl-port 49
username 49
proxy servers 48, 49
proxyset 49
Publishing Utility
system requirements 9
purge schedule
every day 22
only on 22
only on these days 22

R
receiver 54
regex filter type 33
remote-host 52
replicator 48
replicator logging facility 21
required 49
resourcePath 59
retry-delay 55
rights-holder job element 27
rolebase 57
rolename 57
rolesearch 57
rolesubtree 58
root element 41
run-interactive 45

S
scheduler 48
scheduler logging facility 21
security 48
security logging facility 21
self-url 52, 53, 54
serializer 48
Server CGI URL 16
server identity 21
server URL for subscriber 22
server UUID 21
services element 50
showcredit job element 27
shutdown-slaves 54
Site Studio Publishing Utility 5
soap 48
soap.service logging facility 21
soap-server host element 60
src attribute 33
ssl element 49
   admin 50
   enabled 49
   keystorefilename 50
   keystorepass 50
   required 49
ssl-port 49, 52, 53
start 51, 52, 53, 54
start-browser 45
startdate attribute 29
starttime attribute 29
Stellent Site Studio
   publishing a web site 5
stopdate attribute 29
Subscription Client 9
subscription-id job element 26
subscription-state job element 26
success-exit-code attribute 34
sweeper 48
synchronized delivery option 16
syndicator 48
syndicator logging facility 21
syntax
   cmd trigger 35
   http-post trigger 36
   java trigger 37

T

template 48
template logging facility 21
time format 23
timeFormat element 45
tracking-server host element 61

trigger
   sample 38
   trigger type
      cmd 35
      http-post 36
      java 37
   triggers, described 35
type attribute 32

U

ui 48
ui.validation logging facility 21
ui.web logging facility 21
update schedule
   every day 16
   first update at 17
   last update at 17
   manual update 17
   only on 16
   only on these dates 16
   update every 17
URL
   Microsoft Netshow server 26
   RealAudio 26
url job element 25
usage-required job element 27
use-input-as-output attribute 34
user guide
   conventions 8
user-agent job element 26
username 49, 55, 57
username job element 25
userpassword 57
userpattern 58

V

verbose 19, 46
virtualDir 59

W

warning 19, 46
web site
   publishing 5
weekday attribute 29
X
xml 48
xml logging facility 21
xsl filter type, described 34