



Lasso 8.5 Setup Guide

Trademarks

Lasso, Lasso Professional Server, Lasso Studio, Lasso Dynamic Markup Language, LassoScript, Lasso Service, Lasso Connector, Lasso Web Data Engine, and OmniPilot are trademarks of LassoSoft, LLC. All other products mentioned may be trademarks of their respective holders. See *Appendix C: Copyright Notices* for additional details.

Third Party Links

This guide may contain links to third-party Web sites that are not under the control of LassoSoft. LassoSoft is not responsible for the content of any linked site. If you access a third-party Web site mentioned in this guide, then you do so at your own risk. LassoSoft provides these links only as a convenience, and the inclusion of the links does not imply that LassoSoft endorses or accepts any responsibility for the content of those third-party sites.

Copyright

Copyright © 2007 LassoSoft, LLC. This manual may be printed for your personal use. This manual may not be copied, photocopied, reproduced, translated or converted to any electronic or machine-readable form in whole or in part without prior written approval of LassoSoft, LLC. See *Appendix C: Copyright Notices* for additional details.

Date: March 17, 2008

Version: 8.5.5

LassoSoft, LLC
dba OmniPilot Software
P.O. Box 33
Manchester, Washington 98353
U.S.A.

Telephone: (954) 302-3526
Email: info@lassosoft.com
Web Site: <http://www.lassosoft.com>

Contents

- Section I**
 - Lasso Overview 10

- Chapter 1**
 - Introduction 11
 - The Lasso 8.5 Product Line 11
 - Lasso Documentation. 12
 - Lasso 8.5 Setup Guide 13
 - Documentation Conventions 14
 - Learning Resources 15
 - Usage Rights 16

- Chapter 2**
 - About Lasso 8.5. 17
 - Lasso Fundamentals 17
 - Figure 1: Lasso Professional Server Architecture* 18
 - Figure 2: Lasso Architecture and Extensibility* 19
 - Figure 3: Lasso Professional Server Four Phases of Operation* 20
 - What's New in Lasso 8.5 21
 - Lasso 8 Features. 23
 - Lasso 8 Features Overview 26
 - Table 1: Key to Chart Symbols.* 26

Section II	
Installation	32
Chapter 3	
Configuring on Mac OS X	33
System Requirements	33
Installation Instructions	34
<i>Figure 1: Server Initialize Page</i>	36
<i>Figure 2: Server Administration</i>	37
Running Lasso Professional Server	37
Extended Configuration	38
Installation Contents	44
Uninstalling Lasso Professional Server	47
Chapter 4	
Configuring on Windows	49
System Requirements	49
Installation Instructions	50
<i>Figure 1: Initialize Page</i>	53
<i>Figure 2: Server Administration</i>	54
Enabling Auto Recovery	54
Running Lasso Professional Server	54
Extended Configuration	55
Installation Contents	59
Uninstalling Lasso Professional Server	61
Chapter 5	
Configuring on Linux	63
System Requirements	63
Required Packages	64
<i>Table 1: Lasso Service Required Packages</i>	64
<i>Table 2: Lasso Connector for Apache 2 Required Packages</i>	64
Web Browser Requirements	65
Installation Instructions	65
<i>Figure 1: Server Initialize Page</i>	67
<i>Figure 2: Server Administration</i>	68
Running Lasso Professional Server	68
Extended Configuration	70
Installation Contents	72
Uninstalling Lasso Professional Server	74
Chapter 6	
Upgrading	76
Upgrading Lasso Professional 8 or 8.5	76
Overview	77
Upgrading on Mac OS X	78
Upgrading on Windows	79
Upgrading On Linux	81
Upgrading From Earlier Versions	82

Section III	
Server Administration	83
Chapter 7	
Using Server Administration	84
Accessing Server Administration	84
<i>Figure 1: Server Administration</i>	85
Exploring Server Administration	86
<i>Figure 2: Sites Section</i>	86
<i>Figure 3: Utility Section</i>	87
Interface Elements	87
Lasso Server Databases	87
Chapter 8	
Setting Server Preferences	89
Summary	89
<i>Figure 1: Summary Page</i>	89
Serial Numbers	90
<i>Figure 2: Serial Number</i>	90
Administrator Settings	90
<i>Figure 3: Admin Settings Page</i>	91
Tag Settings	91
<i>Figure 4: Tags List Page</i>	92
Software Updates	92
Import and Export Settings	92
<i>Figure 5: Import/Export Page</i>	93
Chart FX	93
<i>Figure 6: Chart FX Page</i>	94
Chapter 9	
Setting up Sites	95
Overview	95
Setting Up Sites	98
<i>Figure 1: Sites List Page</i>	99
<i>Figure 2: Add Site Panel</i>	99
Site Preferences	101
<i>Figure 3: Site Preferences</i>	102
Allowed Files Paths	102
<i>Figure 4: Allowed File Paths</i>	102
Managing Sites	103
Chapter 10	
Server Utilities	105
SQL Browser	105
<i>Figure 1: SQL Browser Page</i>	105
Errors	106
<i>Figure 2: Errors Page</i>	107
<i>Figure 3: Error Setup Page</i>	107
Threads	108
<i>Figure 4: Threads Page</i>	108

Section IV	
Site Administration	109
Chapter 11	
Using Site Administration	110
Accessing Site Administration	110
<i>Figure 1: Site Administration</i>	111
Exploring Site Administration	111
<i>Figure 2: Setup Section</i>	112
<i>Figure 3: Utility Section</i>	112
<i>Figure 4: Support Section</i>	113
Interface Elements	113
Lasso Site Databases	113
Chapter 12	
Setting Site Preferences	115
Summary	115
<i>Figure 1: Summary Page</i>	115
Administrator Settings	116
<i>Figure 2: Admin Page</i>	116
Lasso Settings	116
<i>Figure 3: Lasso Settings Page</i>	117
File Extension Settings	119
<i>Figure 4: File Extensions Page</i>	119
Session Settings	120
<i>Figure 5: Sessions Page</i>	120
Java Settings	123
<i>Figure 6: Java Page</i>	123
Import and Export Settings	124
<i>Figure 7: Import/Export Page</i>	124
Chapter 13	
Setting Up Data Sources	126
Overview	126
<i>Table 1: Lasso Professional Server 8.5 Data Sources</i>	127
FileMaker Pro Data Sources	128
<i>Table 2: Supported Versions of FileMaker Pro's Web Companion</i>	129
<i>Figure 2: FileMaker Pro Hosts Page</i>	130
FileMaker Server Data Sources	132
<i>Figure 3: FileMaker Server Hosts Page</i>	134
JDBC Data Sources	135
<i>Figure 5: JDBC Hosts Page</i>	137
MySQL Data Sources	138
<i>Figure 6: MySQL Hosts Page</i>	139
ODBC Data Sources	141
<i>Figure 7: ODBC Hosts Page</i>	142
OpenBase Data Sources	143
<i>Figure 8: OpenBase Hosts Page</i>	143
Oracle Data Sources	145
<i>Figure 9: Oracle Hosts Page</i>	146
PostgreSQL Data Sources	147

<i>Figure 10: PostgreSQL Hosts Page</i>	148
Spotlight Data Source	149
SQL Server Data Sources	150
<i>Figure 11: SQL Server Hosts Page</i>	151
SQLite Data Sources	153
Other Data Sources	153
Database Preferences	154
<i>Figure 12: Databases Preferences Page</i>	154
Table Preferences	156
<i>Figure 13: Table Preferences Page</i>	156
Field Preferences	157
<i>Figure 14: Field Preferences Page</i>	157

Chapter 14

Setting Up Security 160

Introduction	160
<i>Figure 1: Group Detail</i>	162
<i>Figure 2: Custom Groups</i>	162
Setting Up Groups	164
<i>Figure 3: Groups List Page</i>	165
<i>Figure 4: Add Group Panel</i>	165
Setting Up Users	166
<i>Figure 5: Users List Page</i>	166
<i>Figure 6: Add User Panel</i>	167
<i>Figure 7: Group Administration LassoApp</i>	168
<i>Figure 8: Hosts Page</i>	169
Database Permissions	170
<i>Figure 9: Databases Page</i>	170
Table Permissions	172
<i>Figure 10: Tables Page</i>	172
Field Permissions	173
Tag Permissions	174
<i>Figure 11: Tags Page</i>	174
File Permissions	175
<i>Figure 12: Files Page</i>	175
Web Server Security	177
Record-Level Security	178
Security Tips	179

Chapter 15

URL Rewrite 182

Introduction	182
Rules, Triggers, and Actions	183
Setting Up Rules	187

Chapter 16

Site Utilities 191

SQL Browser	191
<i>Figure 1: SQL Browser Page</i>	192
Email	193
<i>Figure 2: Email Queue Page</i>	194
<i>Figure 3: Email Routes Page</i>	195

Figure 4: Send Email Page	195
Figure 4: Setup Page	196
Events	197
Figure 5: Event Queue Page	198
Figure 6: Schedule Event Page.	198
Errors	200
Figure 7: Errors Page	201
Figure 8: Error Setup Page	201
LassoApps	202
Figure 9: LassoApp Build Page	204
Figure 10: List LassoApps Page	205
Figure 11: LassoApp Settings Page	205
Caches	205
Figure 12: Cache List Page	206
Figure 13: Cache Settings	207

Chapter 17

Building and Browsing Databases 208

Overview	208
Figure 1: Lasso Database Browser	209
Database Browser	209
Figure 2: Settings Sidebar	210
Figure 3: Search Page	210
Figure 4: Search Options.	211
Figure 5: Add Page	212
Figure 6: Add Options	212
Figure 7: List Page.	213
Figure 8: List Options	214
Figure 9: Detail Page	215
Figure 10: Detail Options	216
Figure 11: Update Page	217
Figure 12: Update Options	217
Figure 13: Import Page	219
Figure 14: Export Page	220
Figure 15: SQL Browser	222
Database Builder.	222
Figure 16: Databases Page	222
Figure 17: Create Database Panel.	223
Figure 18: Duplicate Database Panel	223
Figure 19: Tables Page.	224
Figure 20: Create Table Panel	224
Figure 21: Duplicate Table Panel.	225
Figure 22: Fields Page	226
Figure 23: Create Field Panel	227
Figure 24: Indices Page	227
Figure 25: Create Index Panel.	228

Appendix A
Glossary 229

Appendix B
License Agreement..... 235

Appendix C
Copyright Notices..... 237

Appendix D
Index..... 242

Section I

Lasso Overview

This section presents an overview of the Lasso product line, Lasso Professional Server, and the documentation included with it.

- *Chapter 1: Introduction* shows how Lasso Professional Server relates to the other products in the Lasso product line, detail documentation conventions, and includes a discussion of the usage rights for this software.
- *Chapter 2: About Lasso Professional Server* introduces the product including the architecture, extensibility, and mode of operation. Also includes details about what's new in Lasso Professional Server and a feature chart.

For a quick start, see *Section II: Installation* for instructions about how to install Lasso Professional Server on the supported platforms and how to upgrade from a previous version of Lasso.

1

Chapter 1 Introduction

This chapter provides an overview of the Lasso 8.5 product line, documentation, and learning resources available at LassoSoft.

- *The Lasso 8.5 Product Line* introduces the full range of Lasso 8 products and how they relate to Lasso Professional Server.
- *Lasso Documentation* describes all documentation included with Lasso Professional Server.
- *Lasso 8.5 Setup Guide* describes the sections of the book you are reading now.
- *Documentation Conventions* includes information about typographic conventions used within the documentation.
- *Learning Resources* provides pointers to resources where you can learn more about using and programming Lasso Professional Server.
- *Usage Rights* includes important information about usage rights for Lasso Professional Server.

The Lasso 8.5 Product Line

The Lasso product line consists of authoring and serving tools that allow Web designers and Web developers to quickly build and serve powerful data-driven Web sites with maximum productivity and ease. The product line includes Lasso Professional Server for building, serving, and administering data-driven Web sites, and Lasso Studio™ for building and testing data-driven Web sites within a graphical editor.

Lasso's innovative architecture provides an industry first multi-platform, database-independent and open standards approach to delivering database-driven Web sites firmly positioning Lasso technology within the rapidly evolving server-side Web tools market. Lasso technology is used at hundreds of thousands of Web sites worldwide.

Lasso Professional Server

Lasso Professional Server is used for building, serving, and administering data-driven Web sites. It consists of a core Lasso Service application and several Lasso Connectors which establish links to Web servers and data sources. Lasso Professional Server is available for Mac OS X, Red Hat Linux, Windows 2000, Windows XP Professional, and Windows Server 2003. Lasso Professional Server has the following components.

- **Lasso Service** – The core application or Web Data Engine.
- **SQLite** – The internal data source of Lasso Professional Server.
- **Lasso Connector for Apache** – Allows Lasso solutions to run via the built-in Apache Web server on Mac OS X or Red Hat Linux or with Apache 2 on any platform.
- **Lasso Connector for IIS** – Allows Lasso solutions to run via Microsoft's Internet Information Services on Windows XP Professional, Windows Server 2000, or Windows Server 2003.

- **Lasso Connector for WebSTAR V** – Allows Lasso solutions to run via 4D’s WebSTAR V Server Suite on Mac OS X.
- **Lasso Connector for J2EE** – Allows Lasso solutions to run via J2EE compatible servers.
- **Lasso Data Source Connectors** – Allow Lasso to connect to a wide range of data sources including: MySQL, FileMaker Pro 4.x, 5.x, or 6.x, FileMaker Server Advanced 7.x, 8.x, 9.x, FileMaker Server 9.x, PostgreSQL, Oracle, OpenBase, Microsoft SQL Server, and SQLite. In addition data sources can be accessed through ODBC or JDBC. Lasso also provides access to Spotlight on Mac OS X.

Lasso Developer

Lasso Developer is a single-user edition of Lasso that can be used by a single developer to create and test interactive Web sites on their own machine. Lasso Developer has a limitation so only 5 client IP addresses can access it and a per minute transaction limit.

Any installation of Lasso will default to Lasso Developer functionality when it is run without a serial number.

Lasso Studio

Lasso Studio is an optional Web development tool for Lasso. It consists of Lasso Developer plus a set of extensions for either Macromedia Dreamweaver or Adobe GoLive that include drag-and-drop LassoScript objects and various builders and editors. Lasso Studio has the following features.

- **Lasso Connectors for Dreamweaver or GoLive** – Allows Lasso Studio to communicate with Lasso Professional Server.
- **Builders** – Construct data-driven sites, HTML forms, URLs, and inlines within a simple, dialog-driven interface.
- **Objects** – Display Lasso code as icons within the design view of the authoring program.
- **Property Inspectors** – Allow tags to be modified without typing code.
- **Syntax Checking and Conversion** – Allows you to debug pages before uploading them to your Web server and to convert pages from older syntax to the most current syntax.

Lasso Studio for Eclipse

Lasso Studio for Eclipse is a development tool which installs into the Eclipse integrated development environment and provides advanced tools for creating and debugging Lasso pages. Lasso Studio for Eclipse includes an integrated debugger, advanced editing and outlining tools, and more. Solutions built using Lasso Studio are deployed using the full, multi-user version of Lasso Professional Server 8.

Lasso Documentation

The documentation for Lasso Professional Server is divided into several different references available in print and electronic formats. All documentation, with the exception of the Lasso 8 Reference, is included in PDF format inside the Lasso Professional 8/Documentation folder installed on your hard drive. The following manuals and resources are included.

- **Lasso 8.5 Setup Guide** – This is the book you are reading now, and is the first guide to the Lasso 8.5 product line you should read. It includes documentation of the architecture of Lasso, installation and configuration instructions, documentation for the administration interface, and provides an introduction to building and serving Lasso solutions.
- **Lasso 8.5 Language Guide** – The documentation of LassoScript, the language used to access data sources, specify programming logic, and much more. This is the core resource for learning how to build Lasso solutions. Also includes a collection of documentation and sample projects which allow you to extend the capabilities of Lasso in C, C++, Java, and LassoScript.

- **Lasso 8.5 Reference** – Provides detailed documentation of each tag in LassoScript. This is the definitive reference to the language of Lasso 8.5. This reference is provided as a LassoApp installed with Lasso Professional Server, and is also available as an online resource from the LassoSoft Web site.

Additional bound and printed copies of each manual may be purchased at <http://store.lassosoft.com>.

Comments, suggestions, or corrections to the documentation are appreciated and may be sent to documentation@lassosoft.com.

Lasso 8.5 Setup Guide

This book contains the following sections and chapters which detail how to set up, install, configure, and administer Lasso Professional Server and Lasso Developer.

Section I – Lasso Overview

- *Chapter 1: Introduction* introduces the Lasso 8.5 Setup Guide and other documentation.
- *Chapter 2: About Lasso 8.5* introduces the Lasso 8.5 product line and describes the components, features, and architecture of Lasso Professional Server and Lasso Developer.

Section II – Installation

- *Chapter 3: Configuring on Mac OS X* provides step-by-step instructions for installing and configuring Lasso on Mac OS X.
- *Chapter 4: Configuring on Windows* provides step-by-step instructions for installing and configuring Lasso on Windows operating systems.
- *Chapter 5: Configuring on Linux* provides step-by-step instructions for installing and configuring Lasso on Red Hat Linux.
- *Chapter 6: Upgrading* provides step-by-step instructions for upgrading from a previous version of Lasso.

Section III – Server Administration

- *Chapter 7: Using Server Administration* describes accessing and using the Web-based administration interface of Lasso Professional Server and Lasso Developer.
- *Chapter 8: Setting Server Preferences* provides instructions on how to set server preferences for Lasso.
- *Chapter 9: Setting Up Sites* provides an introduction to the multi-site features of Lasso and how to set up multiple sites and site criteria.
- *Chapter 10: Server Utilities* describes how to use Server Administration utilities.

Section IV – Site Administration

- *Chapter 11: Using Site Administration* describes accessing and using the Web-based administration interface of Lasso Professional Server and Lasso Developer.
- *Chapter 12: Setting Site Preferences* provides instructions on how to set site preferences for Lasso.
- *Chapter 13: Setting Up Data Sources* provides instructions on how to configure MySQL, FileMaker Pro, PostgreSQL, Oracle, OpenBase, Microsoft SQL Server, Spotlight, and ODBC or JDBC-compliant data sources. Also contains documentation of the internal SQLite data source in Lasso.
- *Chapter 14: Setting Up Security* contains complete documentation of the security features of Lasso including their theory of operation and step-by-step configuration instructions.
- *Chapter 15: Site Utilities* describes how to use Site Administration utilities, including SQLite tools, email, event scheduling, and error reporting.

- *Chapter 16: Building and Browsing Databases* describes accessing and using the Lasso Database Browser, which is a LassoApp included with Lasso Professional Server. The Lasso Database Browser allows the administrator to interact with and test configured databases, as well as create new databases using a Web-based interface.

Appendices

- *Appendix A: Glossary* contains a list of definitions for technical terms covered in this book.
- *Appendix B: License Agreement* contains the Lasso Professional Server license agreement.
- *Appendix C: Copyright Notices* contains copyright notices for components used in Lasso.

Documentation Conventions

This documentation uses several conventions in order to make finding information easier.

Definitions

Definitions are indicated using a bold, sans-serif type face for the defined word. This makes it easy to find defined terms within a page. Terms are defined the first time they are used.

Cross References

Cross references are indicated by an italicized, sans-serif typeface. For instance, the next section in this chapter is *Customer Support*. When necessary, arrows are used to define a path into a chapter such as *Chapter 1: Introduction > Learning Resources*.

Code

Code is formatted in a narrow, sans-serif font. Code includes HTML tags, Lasso tags, and any text which should be typed into a Lasso page. Code is represented within the body text (e.g., [Field] or <body>), or is specified in its own section of text as follows:

```
[Field: 'Company']
```

Code Results represent the results after code is processed. They are indicated by a black arrow, and will usually be the value that is sent to the client's Web browser. The following text could be the result of the code example above.

→ LassoSoft

Lasso tags always appear in code format exactly as they appear except when referring to a group of related tags. For example, -Encoding... refers to all of the encoding keywords (-EncodeNone, -EncodeHTML, etc.). Also, member tags are always referred to beginning with their data type followed by the member tag symbol. For example, the member tag for specifying a character in a text string is referred to as [String->Get].

File Paths

File Paths can be local or fully-qualified file path on either Windows or Mac OS X and are formatted using the same font as code. File paths are represented within the body text (e.g., /// Applications/Lasso Professional 8/Admin or C:\inetpub\wwwroot), or are specified in their own section of text as follows:

```
C:\Program Files\OmniPilot Software\Lasso Professional Server
```

File paths in Mac OS X contain forward slashes (/), while file paths for Windows contain backward slashes (\). If a partial file path is identical on Mac OS X and Windows, then the file path will be shown once with forward slashes (/).

Procedures

The documentation assumes a task-oriented approach. The contents following a task heading will provide step-by-step instructions for the specific task.

To perform a specific task:

- 1 This is the first step of the task.
- 2 This is the second step of the task.

Notes

Note: Notes are included to call attention to items that are of particular importance or to include comments that may be of interest to select readers. Notes may begin with **Warning**, **Tip**, **Mac OS X Note**, **JDBC Note**, etc. to specify the importance and audience of the note.

Learning Resources

There are many resources available to help you when installing, administering, building or serving data-driven Web sites powered by Lasso Professional Server.

To find answers to questions about Lasso Professional Server:

- 1 **Documentation** – The Lasso documentation should always be your first resource. Check to see if the Lasso 8.5 Setup Guide, Lasso 8.5 Language Guide, or Lasso 8.5 Reference have the answer to your questions. Each PDF guide contains an index and table of contents, and all electronic documents are searchable.
- 2 **LassoSoft Support Central** – A Web resource that allows you to search for updates to the documentation, software updates, and tips about how to best utilize Lasso Professional Server.

<http://support.lassosoft.com>

- 3 **Lasso Talk Forum** – A community of Lasso users who help to answer each other's questions about using Lasso products. If you can't find an answer to your question, then posting to the list will often result in an answer in a short period of time. Information about subscribing to the list and searchable list archives can be found at the following address.

<http://www.listsearch.com/lasso/>

- 4 **Email and Phone Support** – Provided by LassoSoft to qualified customers during normal business hours. Visit the following Web site for more information.

<http://support.lassosoft.com/>

Usage Rights

These are the usage rights for Lasso Professional Server and Lasso Developer. Please consult the Lasso Studio User Guide for important information about the usage rights and license agreements specific to that products.

Lasso Developer

When any installation of Lasso 8.5 is run without a serial number it defaults to Lasso Developer functionality. Lasso Developer can be used for development of Lasso-based solutions and to preview those solutions to clients. Lasso Developer is not licensed for deployment. Please purchase a license for Lasso Professional Server before deploying your solutions.

Lasso Developer has all the features of Lasso Professional Server, but includes a 5 IP address limitation and a limit on the number of connections which Lasso will process per minute.

New Purchase

Your license permits a single copy of Lasso Professional Server to be installed and used on a single computer. While certain components of Lasso Professional Server (e.g. Lasso Web server connectors) may be installed on a separate computer from the core of Lasso Professional Server (Lasso Service), only a single instance of each component is permitted. The license does not permit deployment using Lasso Developer or evaluation versions.

Upgrade Purchase

It is standard industry practice and understood that by upgrading one's software one no longer uses the old version and the license to use and transfer said license ceases. LassoSoft allows for a 45 day grace period during which you may transition your sites from one version of Lasso to another.

If you have upgraded to Lasso Professional Server from any previous version of Lasso, after the grace period you must no longer use that version. Please see the termination provisions in the accompanying license agreement for further details.

Alternatively, you could purchase a new Lasso Professional Server license and not be bound by such upgrade restrictions.

Evaluation Versions

Evaluation versions are provided for one-time 30-day evaluation and initial product testing use. Evaluation versions are not licensed for use for deployment. Please purchase a license for Lasso Professional Server before deploying your solutions.

2

Chapter 2

About Lasso 8.5

Welcome to Lasso Professional Server, the premiere tool for serving and administering data-driven Web sites. This chapter introduces important concepts and information you should know before starting to use Lasso Professional Server and Lasso Developer.

- *Lasso Fundamentals* introduces the architecture and operation of Lasso Professional Server and Lasso Developer.
- *What's New in 8.5* presents the new features in Lasso compared to Lasso Professional 8.
- *Lasso 8.5 Features Overview* provides a chart of all the features of Lasso Professional Server and Lasso Developer with the new features highlighted.

Lasso Fundamentals

Lasso Professional Server (and Lasso Developer) is server-side software that adds a suite of dynamic functionality and administration to your Web server. This functionality empowers you to build and serve just about any dynamic Web application that you need with maximum productivity and ease.

Lasso works by using a simple tag-based markup language (LassoScript), which can be embedded in Web pages and scripts residing on your Web server. The details of scripting and programming in LassoScript are covered in the Lasso 8.5 Language Guide also included with this product.

By default, Lasso Professional Server is designed to run on the most prevalent modern Web server platforms with the most popular Web serving applications. In addition, Lasso's extensibility allows Web Server Connectors to be authored for any Web server for which default connectivity is not provided.

Macintosh

On Mac OS X, Lasso Professional Server is implemented as a Universal binary application and framework which interfaces with the default Apache Web server provided with Mac OS X. Lasso will run natively on PowerPC or Intel-based Macintosh computers. Lasso Professional Server will also work with WebSTAR V, Apache 2, or J2EE compatible servers on Mac OS X or Mac OS X Server.

Windows

Lasso Professional Server runs as a service on Windows 2000, Windows XP Professional, and Windows Server 2003, and interoperates with the included Internet Information Services (IIS) Web services. Lasso Professional Server will also work with Apache 2 or J2EE compatible servers on Windows.

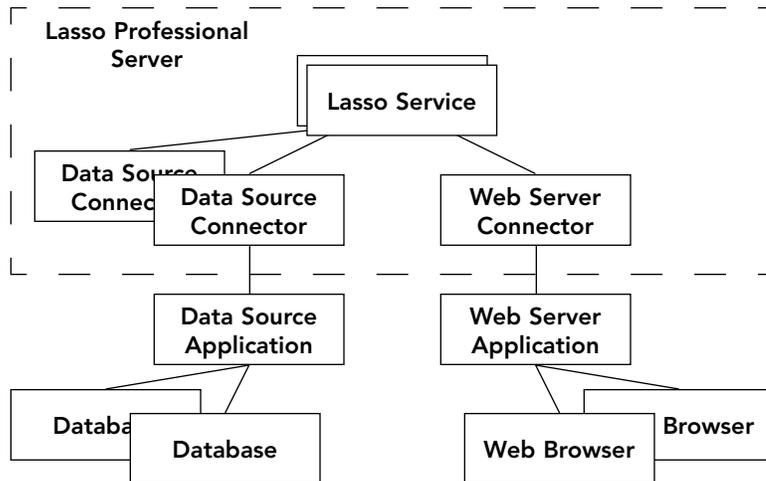
Red Hat Linux

Lasso Professional Server runs as a daemon on Red Hat Linux, and interoperates with the included Apache Web server. Lasso Professional Server will also work with Apache 2 or J2EE compatible servers on Red Hat Linux.

Architecture

Lasso Professional Server's architecture is illustrated in *Figure 1: Lasso Professional Server Architecture*. The core of Lasso Professional Server is Lasso Service (i.e. the Web Data Engine). One or more Lasso data source connectors provide connectivity to data source applications which can be hosts to many databases. Likewise, a single Lasso Web server connector provides connectivity to a Web server application which serves data to many Web browsers.

Figure 1: Lasso Professional Server Architecture



Using multi-site features, multiple Lasso Service processes may be running at one time. The master Lasso Service routes incoming Web requests to the appropriate child Lasso Service based on site criteria. See the chapter on *Setting Up Sites* for more information.

To serve a page from a data-driven Web site the following steps occur:

- 1 A request originates in a Web browser as a result of a client following a link or selecting a URL (e.g., `http://www.example.com/default.lasso`).
- 2 The request is sent to the Web server application which is configured to send all URLs that contain the `.lasso` extension to the Lasso Web server connector.
- 3 The request is sent through the Lasso Web server connector to Lasso Service along with the Lasso page requested in the URL (e.g., `default.lasso`).
- 4 If multiple sites have been configured in Lasso then the site criteria are used to route the incoming request to the appropriate site. This determines what data sources and security settings will apply when the page is processed.
- 5 The Lasso page is processed, interpreting the embedded Lasso tags.
- 6 If required, Lasso Service sends one or more queries through one of the Lasso data source connectors to a database hosted by a data source application.
- 7 The database query results are sent back to Lasso Service and merged with the presentation and logic elements contained in the requested Lasso page.
- 8 The formatted results are sent back to the Web server application via Lasso Web server connector.
- 9 The Web server application sends the completed page back to the requesting Web browser.

Although this seems like a lot of steps for a simple URL request, Lasso can handle many thousands of requests per hour, each with a unique combination of concurrent database requests and programmatic functions.

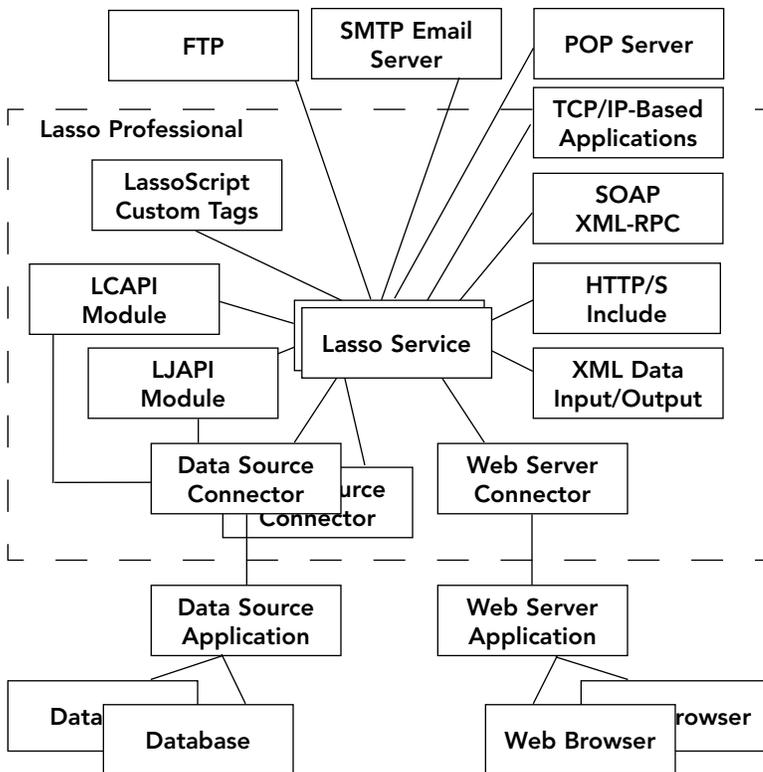
Extensibility

The full wealth of connectivity and extensibility features available in Lasso Professional Server are shown in *Figure 2: Lasso Architecture and Extensibility*. Lasso Service is a hub that bridges the gaps between many different technologies in order to provide you with the flexibility to create whatever type of data-driven site you need.

Lasso Professional Server can connect to other Web Servers and SMTP email servers through a built-in TCP/IP client. It can include content from other Web servers through HTTP include functionality. XML data can be both served by Lasso as well as received from other Web servers via HTTP includes and built-in XML Web services.

Lasso Professional Server can be extended through LassoScript custom tags and types that can be defined within a Lasso page or collected into a library for re-use. New tags can be created by third parties in Java and distributed as an LJAPI tag module. New tags or even new Lasso data source connectors can be created in C/ C++ and distributed as LCAPI modules.

Figure 2: Lasso Architecture and Extensibility

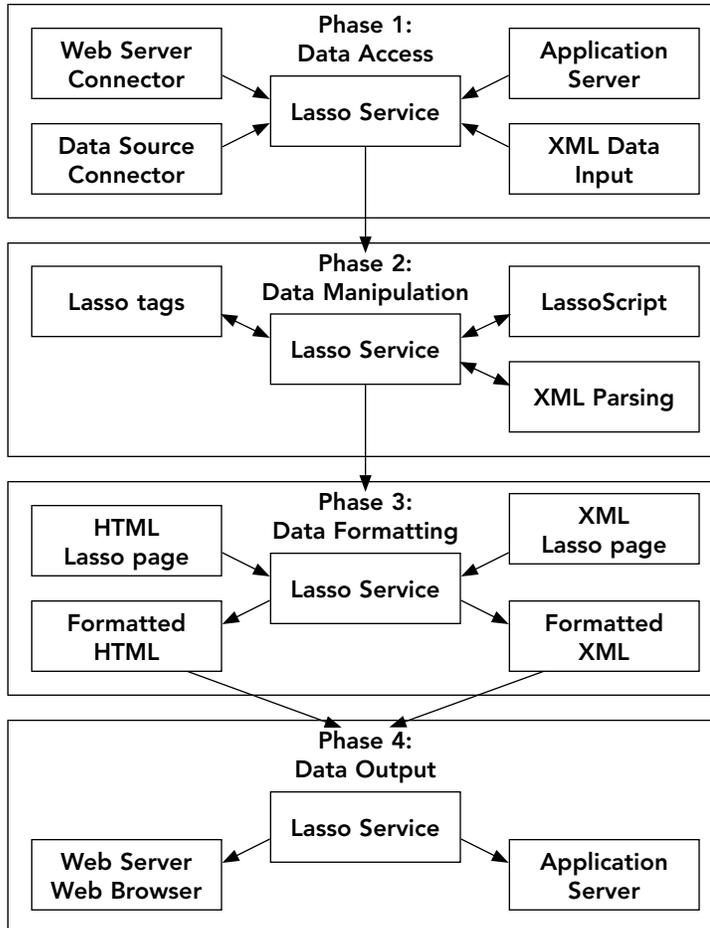


Each of the technologies in the figure are discussed in more detail in later chapters of this Lasso 8.5 Setup Guide or in the Lasso 8.5 Language Guide.

Four Phases of Operation

Lasso Professional Server is a Web Data Engine that operates in four phases for each incoming URL or HTML form submission. These four phases are: Data Access, Data Manipulation, Data Formatting, and Data Output.

Figure 3: Lasso Professional Server Four Phases of Operation



- 1 Data Access** – The Data Access phase includes gathering data from a Lasso Web server connector (the URL request or HTML form input), Lasso data source connectors (results of database queries), and external application servers and XML data sources (through the HTTP include mechanism).
- 2 Data Manipulation** – The Data Manipulation phase is where Lasso tags, LassoScripts, and XML parsing routines are used to modify the data from the previous phase.
- 3 Data Formatting** – In the Data Formatting phase, the results from the manipulated data are merged with the style and graphics in your Lasso page template and formatted for output as HTML, XML or other formats depending upon the target client.
- 4 Data Output** – Finally, the Data Output phase is when the data is actually delivered back to the client which initiated the request. The data is sent as HTML or XML data. The Web server application typically handles follow-up requests calling on Lasso Service as needed.

What's New in Lasso 8.5

Lasso 8.5 supports the same syntax as Lasso 8, but offers a compelling set of new data sources and features which make building dynamic Web sites even easier. Lasso 8.5 includes the following new features.

Backward Compatibility

All solutions written using preferred LassoScript 7 or 8 syntax should run in Lasso Professional Server without modification. See *Section IV: Upgrading* in the Lasso 8.5 Language Guide for details about changes to Lasso.

Lasso Developer

When any installation of Lasso 8.5 is run without a serial number it defaults to Lasso Developer functionality. Lasso Developer can be used for development of Lasso-based solutions and to preview those solutions to clients. Lasso Developer has all the features of Lasso Professional Server, but includes a 5 IP address limitation and a limit on the number of connections which Lasso will process per minute. Lasso Developer is not licensed for deployment of solutions.

AJAX Support

AJAX (Asynchronous JavaScript and XML) support is provided through the new LJAX tags and JavaScript library. AJAX allows a Web site to be updated asynchronously providing a more rich user experience.

OS Process Type

The [OS_Process] type allows access to native processes running on the same machine as Lasso. This type can be used to access the shell on all three platforms, AppleScript on Mac OS X, or to access machine resources such as printers.

URL Rewrite

Built-in URL rewrite tool allows Lasso to accept incoming URLs and map them to specific Lasso pages, redirect to another server, etc.

Data Sources

Lasso 8.5 includes many new native data source connectors. In addition to the MySQL and FileMaker connectors provided in Lasso 8, the following data sources are now officially supported:

- **PostgreSQL** – Access this popular open source data source.
- **Microsoft SQL Server** – Access Microsoft's flagship data source.
- **Oracle** – Access one of the most popular enterprise data sources.
- **OpenBase** – Access this fast and powerful data source.

Lasso 8.5 includes support for any data source which provides a compatible ODBC driver, in addition to the JDBC access provided in Lasso 8.

- **ODBC** – Access any data source which provides a compatible ODBC driver.

In addition, Lasso 8.5 includes a data source connector for Mac OS X's built-in search technology.

- **Spotlight** – On Mac OS X only, the Spotlight data source allows file system searches to be performed from within Lasso.

Inline Enhancements

- **Dynamic Hosts** – Data source hosts can now be specified directly within an inline. This allows data sources to be configured quickly without using the site administration tools or Lasso's built-in security.
- **Key Fields** – Key fields and key values can now be specified using complex search criteria with the new `-Key` parameter. This allows groups of records to be updated or deleted using a standard inline.
- **Prepared Statements** – Lasso now supports prepared statements to help optimize database access. A prepared statement allows a database operation to be set up and then used multiple times.
- **Multiple Result Sets** – Multiple result sets from an inline can now be accessed individually. This allows batch operations to be issued as single transactions and the results of each operation to be examined individually.
- **Quick Operators** – Inline operators can now be abbreviated `-BW`, `-EQ`, `-LT`, `-GTE`, etc.

Regular Expressions

New regular expression data type allows for faster and more powerful string manipulations including the ability to interactively modify each replacement value before inserting it into the string.

LassoScript Enhancements

- **File Encoding** – The `[File]` object has been enhanced with additional encoding parameters which make it easy to read or write data using any character encoding. The `[Net]` object is similarly enhanced.
- **[Encode_URL]** – URL encoding has been enhanced to use the current page character set rather than defaulting to UTF-8 encoding.
- **Page Environment** – New tags provide access to previously hidden variables which allow the page environment to be inspected and modified including the current text which will be returned to the browser and the current HTTP header.
- **Include** – A new `[Include_Once]` tag tells Lasso to process a file only once even if it is included multiple times. New tag returns the name of the current processing page whether it is an include or a Lasso page.
- **Namespaces** – All tags are now looked up using a unified namespace mechanism whether they were defined with the `-Namespace` parameter or not.
- **Private Members** – Custom data types now support private members which can only be accessed from within the data type itself.
- **Big Integer Type** – `[BigInt]` type allows math operations to be performed on very large integers without loss of precision.

iCalendar Support

Lasso can parse, merge, create, and output calendars and calendar items in the industry standard iCalendar format.

Email Enhancements

- **Direct Send** – Email messages are now sent directly to remote SMTP servers. Lasso no longer requires a local SMTP server to relay messages, but can still utilize one if required.
- **Attachments** – Email attachments can now be generated programmatically.
- **Inline Images** – Images can now be embedded properly within HTML messages so that they display inline and not as attachments.
- **Mail Merge** – Messages can now be multiply addressed with mail merge parameters which are applied just before the message is queued. This allows a single email message to be sent to many recipients with customizations.

- **SSL Encryption** – Both SMTP and POP now support SSL connections for encrypted communication with email servers.
- **Authentication** – Both SMTP and POP support CRAM-MD5 and DIGEST-MD5 authentication for compatibility with a wider range of email servers.
- **Performance** – The performance of encoding and decoding messages has been improved.

Administration Enhancements

- **Threads** – Lasso now allows the running threads to be listed within the administration interface. Individual threads can be aborted or set to higher or lower priority.
- **Interface** – The administration interface now makes better use of screen real estate. The interface is resizable and the tabs and footer are fixed within the window. In addition, many sections have been rewritten to present more information within the browser window without sacrificing functionality.
- **SMTP Queue** – The email queue has been enhanced with better error reporting and information about messages which are being sent directly to one or more SMTP servers.
- **Event Scheduler** – The event scheduler has been enhanced with error reporting and with the ability to edit scheduled events.
- **Preferences** – Many characteristics of the administration interface including auto-refresh and listing length can now be configured.

New Tags

For a list of all new Lasso tags in Lasso 8.5, see the Lasso 8.5 Reference hosted at the following address and perform a search on the Lasso 8.5 tag set.

<http://reference.lassosoft.com>

Documentation

The documentation for Lasso Professional Server has been enhanced. The Lasso 8.5 Setup Guide has been updated with information about the new data sources and other features. The Lasso 8.5 Language Guide has been updated with information about new features. The Getting Started Guide has been updated with information about the included Examples Pack. The interface of the Lasso 8.5 Reference has been improved.

Lasso 8 Features

The following features were new in Lasso 8 and are supported by Lasso 8.5. These features are listed here for those users who are upgrading from Lasso 7 or earlier.

Embedded SQLite

Lasso's preferences and security settings are stored using an embedded SQLite data source. SQLite provides greater speed and architectural flexibility. The Lasso Settings Migrator can be used to migrate solution-specific databases from Lasso MySQL to an external MySQL installation.

Multi-Site

Multiple Lasso Service processes can be run simultaneously in order to provide custom configurations for different clients or Web hosts. Each site process has its own data sources, modules, global variables, and LassoApps. Code in one site cannot access any aspect of another site. If a site crashes it will be brought back up automatically without affecting other sites.

Data Sources

- **FileMaker Server** – New data source connector provides communication with FileMaker Server 7 or 8 Advanced and FileMaker Server 9 through the new XML interface.
- **JDBC 3** – Advanced JDBC 3 support provides enhanced connection pooling, better driver selection within Site Administration, and compatibility with JDBC 1 and JDBC 2 drivers.

POP Support

Lasso can retrieve messages from POP email servers. This allows developers to create list archive solutions entirely within Lasso or to create Web-based email solutions. Flexible POP implementation includes support for getting information about waiting messages, fetching message headers, leaving messages on server, parsing MIME encoded messages, processing attachments, and more.

SMTP Support

Improved SMTP implementation includes support for multiple outgoing email queues, sending multiple messages per connection, CRAM-MD5 authentication for SMTP AUTH, built-in support for HTML email, and better attachment handling.

SOAP

Built-in SOAP client allows remote SOAP methods to be called as custom tags within Lasso. Includes support for XSD and WSDL to provide automatic translation of SOAP data types to and from built-in LassoScript data types. In addition, Lasso Professional Server includes full support for serving SOAP methods and XML-RPC support.

Enterprise Java Support

- **J2EE Connector** – Allows Lasso Service to be accessed through J2EE Java Servlet compatible servers such as Tomcat. Lasso can now be accessed as a servlet through Java enterprise servers.
- **JavaBeans** – Lasso Professional Server can access JavaBeans through an easy to use interface. This allows enterprise JavaBeans to be used within Lasso and provides an easy way to extend Lasso with Java functionality.

Compound Data Types

Lasso Professional Server includes a collection of new compound data types that make processing structured data easier and faster. Lasso Professional Server's compound data types include arrays, maps, pairs, sets, queues, stacks, priority queues, and trees.

Inline Enhancements

MySQL inlines support the GROUP BY parameter using `-GroupBy`. A log-level option on each inline provides control over whether the SQL statement generated by the inline is logged or not. A new `[Action_Statement]` tag also provides access to the generated SQL statement.

On-Demand Tag Libraries and LassoApps

Libraries of custom tags and LassoApps can be loaded on-demand from within the Lasso Professional Server application folder. This allows custom tags to be provided server-wide without restarting Lasso Service. LassoApps-based solutions can be served and updated without modifying any files in the Web server root.

Recursion Protection

Recursion protection has been extended to custom tag calls. Unlimited recursion in custom tags will now be caught and reported. This protection is in addition to Lasso's protection for recursive includes and rogue LassoScript page processes.

Digest Authentication

Lasso Professional Server supports digest authentication in addition to basic authentication for Web clients. Digest authentication is more secure than basic authentication since it uses one-way hash encryption on all transmitted password. Digest authentication also provides better support for logging users out of a Web site without forcing them to quit their browser.

Parentheses Syntax

New syntax style allows tags to be specified as the tag name with parameters included in parentheses [Tag_Name(Parameters)]. This new syntax style is provided in addition to the existing colon syntax supported by prior versions of Lasso [Tag_Name: Parameters]. This new parentheses syntax style is less ambiguous than colon syntax and may be more comfortable for programmers who are familiar with other programming languages.

Namespaces

Lasso Professional Server introduces the concept of namespaces which form the foundation for the on-demand tag loading and SOAP features of Lasso Professional Server.

File Upload Enhancements

New tags provide more information about uploaded files and make it easy to copy uploaded files to a target directory. Flexible file upload filters allow file sizes and extensions to be checked automatically.

Administration Enhancements

Lasso Professional Server features separate Server Administration and Site Administration LassoApps which are served on-demand from within the Lasso Professional Server application folder. The Database Browser has been enhanced with SQLite support.

LassoScript Enhancements

- **Image Tags** – Lasso Professional Server has improved image manipulation tags using ImageMagick 6. New tags allow images to be created from individual color files and for color histograms to be created.
- **Bytes Support** – New options on the file, network, and [Include_URL] tags provide enhanced support for binary data.
- **Retarget Symbol** – The retarget symbol & allows a member tag which modifies a data type directly to be called and then the root data type to be returned for further processing. This allows multiple member tags to be called on the same root data type in a single expression.
- **Custom Tag Encoding** – Encoding can now be controlled when custom tags are defined.
- **LassoScript Data Source Connectors** – New data sources can be coded entirely in LassoScript for access to data sources through any of Lasso's built-in file or networking tags.
- **BlowFish and Encryption** – Improved BlowFish algorithm provides better interoperability with other Web application servers. New encryption tags provide access to many popular encryption, digest, and encoding methods.
- **Page Hooks** – Allow code to be executed before each page load on a site or for code to be executed immediately after a page load.

- **Data Type Prototypes** – Improves the performance of creating custom data types in LP8 by allowing a prototype instance to be created which is copied when new instances must be created.

Lasso 8 Features Overview

The following provides a list of features in Lasso Professional Server. Please consult this manual or the Lasso 8.5 Language Guide for details on each of these features.

Table 1: Key to Chart Symbols

(8.5)	New Feature in Lasso Professional.	(E)	Enhanced from Prior Versions.
(M)	Mac OS X Only.	(T)	Available through Third Parties.
(W)	Windows Only.	(R)	Licensing Restrictions Apply.
(S)	Available in Lasso Studio.		

Upgrading Note: All solutions written using preferred LassoScript 7 or 8syntax will operate in Lasso Professional Server. For a list of tags and features that have been altered, see *Section IV: Upgrading* in the Lasso 8.5 Language Guide.

Operating Systems and Web Servers

- Operates on Windows 2000, Windows XP Professional, or Windows Server 2003 as service application. (W)
- Operates on Mac OS X or Mac OS X Server as native Darwin application. (M)
- Universal binary for native support of both PowerPC and Intel-Based Macintosh computers. (8.5) (M)
- Supports Internet Information Server (IIS) version 5 or 6 as ISAPI plug-in. (W)
- Supports included Apache 1.3 on Mac OS X as Apache module. (M)
- Supports Apache 2.0 as Apache module.
- Supports WebSTAR V on Mac OS X as WebSTAR API plug-in. (M)

Client Device Support

- Supports industry standard Web browsers.
- Supports industry standard wireless devices such as cell phones and Personal Digital Assistants (PDAs).
- Supports digest authentication and basic authentication.

Data Source Integration

- Includes embedded SQLite datasource.
- Supports external MySQL databases via Lasso Connector for MySQL.
- Supports external OpenBase databases via Lasso Connector for OpenBase.
- Supports external Oracle databases via Lasso Connector for Oracle.
- Supports external PostgreSQL databases via Lasso Connector for PostgreSQL.
- Supports external Microsoft SQL Server databases via Lasso Connector for SQL Server.
- Supports direct connection to FileMaker Pro 4, FileMaker Pro 5, and FileMaker Pro 6 Web Companion enabled databases via Lasso Connector for FileMaker Pro.
- Supports direct connection to FileMaker Server 7 or 8 Advanced and FileMaker Server 9 via Lasso Connector for FileMaker Server.

- Includes integrated JDBC 3 data source support to allow connections to JDBC-compatible data sources with enhanced driver selection and connection pooling.
- Support for ODBC data sources via Lasso Connector for ODBC.
- Support for Spotlight on Mac OS X 10.4/10.5 via Lasso Connector for Spotlight. (M)
- Provide built-in SQL Query Browser for building, browsing, and editing SQL-based databases.

Authoring Tools

- Allows Lasso code to be authored from within any text or HTML editor.
- Provides graphical development environment within Adobe GoLive CS and Macromedia Dreamweaver MX 2004 for building Lasso-based sites. (S)
- Available glossary of Lasso tags for drag-and-drop insertion within Macromedia HomeSite. (W) (T)

Architecture

- Operates via a combination of Lasso Connector plug-ins for Web servers and a Lasso Service core application operating on the same machine or distributed across two machines. (R)
- Multi-site features allow multiple Lasso Service processes to be created. Site criteria determine to which site incoming URL requests are routed. If a site crashes it can be restarted without affecting any other sites.
- Database connectivity provided via Lasso data source connector modules. (E)
- Supports multiple Lasso server configurations with shared session data. (R)
- Supports remote installation on Mac OS X. (M)
- Supports multiple Lasso server configurations with shared session data. (R)
- URL Rewrite – Built-in URL rewrite technology allows dynamic URLs to be created entirely within Lasso (8.5).

Lasso Service

- Optimized parser designed to support multiple syntax styles including colon syntax and parentheses syntax.
- Multi-threaded processing.
- Start/stop individual sites irrespective of Web server status.
- Dynamic activation/deactivation of loaded modules per site.
- Customizable console window for controlling what process information is displayed.
- Provides information about running threads and allows threads to be prioritized. (8.5)

Open Source

The following Lasso Connectors are provided with full source code for developer enhancement and customization and to serve as a model for the creation of third party Lasso Connectors for other Web servers and data sources.

- Lasso Connector for MySQL. (W) (M) (L)
- Lasso Connector for JDBC. (W) (M) (L)
- Lasso Connector for Apache 1.3. (M) (L)
- Lasso Connector for Apache 2.0. (W) (M) (L)
- Lasso Connector for IIS. (W)
- Lasso Connector for WebSTAR V. (M)
- Lasso Connector for J2EE.

Languages and Methodology

- Web database connectivity and server-side programming accomplished via an HTML-like tag-based programming language called Lasso Dynamic Markup Language (LassoScript).
- Lasso pages containing HTML, XML, and LassoScript can be stored on a local or remote disk, or within a local or remote database.
- Lasso pages are easily edited within HTML editors using various LassoScript page builders and drag-and-drop LassoScript objects. (S)
- Supports multiple methods of coding LassoScript including LassoScript and traditional square bracket syntax.
- Supports multiple methods of calling Lasso including inlines, URLs, and HTML forms.
- Unicode and UTF-8 encoding are used by default.

Compiled Applications

- Lasso pages and images can be compiled into a LassoApp application file.
- LassoApps can be safely and easily distributed to other users of Lasso Professional Server in order to sell or share Lasso solutions.
- Currently served LassoApps can be listed and managed from within Site Administration.
- LassoApps can be served on-demand from within the Lasso Professional application folder.

Security

- Built-in security engine provides consistent security implementation across platforms.
- Secures data source types, databases, tables/layouts, fields/columns, and records/rows.
- Enable/disable individual tags and categories of tags.
- Provides action filters to protect, validate, or facilitate personalized access to files and data.
- Supports users and groups with unique sets of permissions.
- Allows programmatic manipulation of security via administrative Lasso tags.
- Encrypted storage of usernames and passwords.
- Securely processes and protects files based upon file suffix.
- Enables encrypted transmission of data.
- Import/export security settings.
- Provides built-in, configurable protection against recursion and infinite loop crashes.

Administration

- Remote Web browser based administration. (E)
- Included SQL Query Browser with customization controls for how detail pages should be shown for each database/table.
- Email queue provides details about email which is waiting to be sent and the status of email which has been sent.
- Scheduling queue provides details about events which have been scheduled to execute.
- Enhanced error logging features with options to control what information is displayed.
- Reserves single global administrator for global control of all settings.
- LassoApp Builder allows custom applications to be compiled for deployment with Lasso Professional Server.
- Allow recursion limits and page timeouts to be set.

Programming

- Program database queries inline within Lasso pages.
- Share database connections via nested inlines.
- Built-in data type support including context sensitive member tags.
- Advanced data types including Arrays, Maps, Pairs, Queues, Sets, Stacks, and Trees for structured data storage and manipulation.
- Object-oriented programming features including inheritance and overloading of comparisons and assignments.
- Built-in session management allow clients to be remembered automatically. Includes support for session cookies.
- Specify math and string expressions within tag parameters.
- Perform Unicode string operations for localization or internationalization of data.
- Encrypt or encode data for secure transmission.
- Apply post-processing operations to the entire Lasso page before it is served.
- Provides local custom tag variables, global page variables, and session variables.
- Advanced string manipulation functions including support for regular expressions.
- Advanced error controls including protect/fail blocks and debugging.
- Runtime error reporting for easy debugging.
- Parse XML text, generate XML data, handle XML-RPC requests.
- Communicate with and provide SOAP XML Web services.
- Create, serve, and edit PDF documents.
- Edit and manipulate image files on the fly.
- Serve images and multimedia files.
- Built-in scheduling facility to automatically trigger scheduled events.
- Send email to any SMTP server. Supports multiple email queues and reuse of connections to send multiple emails at once.
- Download email from POP servers.
- Conditional logic tags dynamically represent data based upon user entered or hard-coded criteria.
- Programmatically control network connections and perform file operations via an object-oriented methodology.
- File management tools allow for manipulation of local files and for remote files to be uploaded.
- HTTP access to remote servers including remote form submissions and server-side includes of remote Web pages.
- Automatically PUT and GET files via FTP.
- Automatically GET files via HTTP.
- Communicate with HTTPS servers via SSL.
- Name inline result sets for later retrieval.
- Encoding controls for accurate display of data in Web browsers and other clients. Specify encoding at page, section, or tag level.
- Log data to multiple customized log files or to databases.
- Set HTTP headers and MIME information.
- Set and retrieve cookies.
- Use tokens and URLs to pass data from page to page.
- Advanced pipe, semaphore, and reference functions.

- Advanced integer and decimal math operations.
- Advanced date/time calculations.

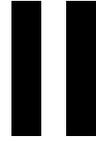
Extensibility

- Custom LassoScript substitution tags and custom LassoScript data types can be defined using LassoScript.
- Custom LassoScript container tags can be defined using LassoScript, LJAPI, or LCAPI.
- Custom data source connectors can be created entirely in LassoScript.
- Custom data types can be created entirely in LCAPI including support for custom member tags and symbol overloading.
- Values of any data type can be processed and returned by LCAPI-based tags.
- Lasso C Application Programming Interface (LCAPI) enables the creation of Lasso data source connector modules, Lasso tags and extended functionality via the C/C++ programming language.
- Lasso Java Application Programming Interface (LJAPI) enables the creation of Lasso tags and extended functionality via the Java programming language.
- Lasso Connector Protocol (LCP) enables the creation of Lasso Web server connectors.
- Supports XML for extensibility via cross-application and cross-server data interchange.
- Functions as a robust TCP/IP client for the creation of custom TCP/IP applications.

Industry Standards

- AJAX – Supports dynamic update of Web pages through asynchronous JavaScript. (8.5)
- CGI – Supports cross server data exchange with other Web application servers.
- DNS – Supports looking up DNS records including IP addresses, reverse lookups, and MX lookups. (8.5)
- File Upload – Supports multi-part HTML forms and file upload.
- FTP – Supports File Transfer Protocol.
- Java – Provides extensibility via Java.
- JDBC – Supports JDBC-compliant databases.
- HTML – Outputs data formatted in HTML for display in HTML compliant devices such as Web browsers.
- HTTP – Supports HTTP 1.1.
- HTTPS – Supports HTTPS and SSL.
- iCalendar – Supports parsing and creating iCalendar documents. (8.5)
- LDAP – Supports the lightweight directory access protocol for security and information lookups. (8.5)
- ODBC – Supports ODBC-compliant databases. (8.5)
- PDF – Outputs data formatted in Portable Document Format for display in PDF-compliant devices.
- SMTP – Supports email forwarding to SMTP email servers. Includes support for CRAM-MD5 and DIGEST-MD5 (8.5) authentication and SSL encryption (8.5).
- POP – Supports downloading email from POP servers. Includes support for CRAM-MD5 (8.5) and DIGEST-MD5 (8.5) authentication and SSL encryption (8.5).
- SOAP/WSDL – Provides ability to use and serve SOAP XML Web services.
- SQL – Supports Structured Query Language.
- TCP/IP – Built-in TCP/IP client enables interoperability with SMTP email servers, remote Web servers, and other application servers.
- Unicode – Provides support of internationalization and localization of data.
- WAP – Interoperates with wireless devices via WAP.
- WML – Outputs data formatted in WML for display in WML compliant wireless devices.

- XML – Provides cross-application data exchange via XML.
- XML-RPC – Provides ability to output XML data and handle XML-RPC requests.



Section II

Installation

This section describes how to install and configure Lasso Professional Server and how to upgrade from previous versions of Lasso Professional Server.

- *Chapter 3: Configuring on Mac OS X* describes how to install and configure Lasso Professional Server on Mac OS X. Also includes details about all the files that are installed with Lasso Professional Server and instructions for uninstalling.
- *Chapter 4: Configuring on Windows* describes how to install and configure Lasso Professional Server on Windows. Also includes details about all the files that are installed with Lasso Professional Server and instructions for uninstalling.
- *Chapter 5: Configuring on Linux* describes how to install and configure Lasso Professional Server on Red Hat Enterprise Linux. Also includes details about all the files that are installed with Lasso Professional Server and instructions for uninstalling.
- *Chapter 6: Upgrading* describes how to upgrade from an earlier version of Lasso Professional using the Lasso Settings Migrator.

3

Chapter 3

Configuring on Mac OS X

This chapter contains the installation and configuration instructions for Lasso Professional Server on Mac OS X. It also contains information on Lasso Professional Server components and how they can be manually installed or modified.

- *System Requirements* lists the minimum system requirements for Lasso Professional Server.
- *Web Browser Requirements* lists the Web browser requirements for administering Lasso Professional Server.
- *Installation Instructions* includes step-by-step instructions for installing Lasso Professional Server and establishing a basic setup.
- *Running Lasso Professional Server* describes how to start and stop Lasso Service, and how to check if it is running properly.
- *Extended Configuration* describes how to manually configure Apache for Lasso Professional Server, and how to set up a Lasso Professional Server distributed architecture.
- *Installation Contents* lists the files installed with Lasso Professional Server.
- *Uninstalling Lasso Professional Server* includes step-by-step instructions for removing Lasso Professional Server from your system.

System Requirements

Lasso Professional Server will run on systems which meet the minimum requirements listed below. Although Lasso Professional Server may run on machines which do not meet these requirements, those installations are not supported.

- Macintosh computer with Intel, or PowerPC G5, G4, or G3 processor.
- 512 MB of RAM. More recommended. (Requires an additional 64 MB of available RAM for each additional site. See the *Setting Up Sites* chapter for more information.)
- Standard installation of Mac OS X 10.5, 10.4, or 10.3.9 or Mac OS X Server 10.5, 10.4, or 10.3.9 with an HFS+ formatted hard drive with BSD subsystem option. UFS systems are not supported.
- Mac OS X build of Apache Web Server 1.3, 2.0, or 2.2 included with Mac OS X 10.5, 10.4, or 10.3.9. Or, the latest version of WebSTAR V.
- X11 for Mac OS X must be installed for the [image] tags in Lasso Professional Server to function. For information on obtaining and installing X11, see the Mac OS X install DVD or the following URL:

<http://www.apple.com/macosx/x11/>

Certification Note: Lasso Professional Server is certified to work with default installations of the requirements described above. Modified versions of Apache and customized `httpd.conf` configuration files may work with Lasso Professional Server, but are not officially supported.

Web Browser Requirements

This section describes the basic Web browser requirements needed to successfully set up and administer Lasso Professional Server.

- Safari 2.0 or higher, FireFox, or compatible browser.
- Javascript enabled.
- Cookies enabled.
- Cascading Style Sheets enabled.

If these criteria are not met, then a browser check dialog box will be shown when trying to access Server or Site Administration. You may proceed into the interface without meeting all these criteria, however some elements may not work properly. It is highly recommended that a browser that meets the listed requirements be used.

Installation Instructions

This section provides installation instructions for Lasso Professional Server and its required components and describes how to initialize Server Administration for the first run. For best results, please perform the procedures in the following sections in the order they are presented.

- *Installing X11* describes how to install X11, which is required prior to installing Lasso Professional Server on Mac OS X.
- *Installing Lasso Professional Server* describes how to install Lasso Professional Server.
- *Initialization* describes the initialization of Lasso Professional Server for running the first time.

Important: Instructions for upgrading from Lasso Professional 5, 6, or 7 are included in the *Upgrading* chapter in this section.

Installing X11

X11 (X Window System version 11) for Mac OS X must be installed in order for the image manipulation features of Lasso Professional Server to work (implemented via the [Image] tags). X11 can be optionally installed as part of Mac OS X 10.4/10.5. It is recommended that X11 be installed prior to installing Lasso Professional Server. If X11 is not installed, errors will be logged to Lasso's error log whenever Lasso Service is started and errors will be returned whenever the [Image] tags are used.

To install X11 for Mac OS X:

- Mac OS X 10.4/10.5 users should install X11 from the installation DVD.
- Mac OS X 10.3.9 users can download the X11 installer at the following address:

<http://www.apple.com/support/downloads/x11formacosx.html>

Double-click the X11 installer application, and then follow the instructions to install X11.

Installing Lasso Professional Server

Use the Lasso Professional Server Installer application (Lasso Pro 8.5 Installer.mpkg) to ensure that all of the Lasso Professional Server files are installed in the proper location with the proper permissions. The Lasso Professional Server installer applications supports several different Web server configurations on Mac OS X:

- **Apache** – The default Web server included with Mac OS X and Mac OS X Server. Also referred to as Personal Web Sharing. This will automatically install into Apache 1.3 on Mac OS X 10.4 and lower and on Apache 2.2 in Mac OS X 10.5.
- **Complete Apache 2** – Popular Apache 2 binary distribution for Mac OS X. For more information, see <http://serverlogistics.com/apache2.php>.

- **iAssist Apache 2** – Popular Apache 2 binary distribution for Mac OS X. For more information, see <http://www.maxum.com/iAssist/WebServices.html>.
- **Mac OS X Server Apache 2** – Apache 2 distribution included as an optional installation with Mac OS X Server 10.4 and lower. Note that Mac OS X Server 10.5 users should install the default option.
- **WebSTAR V** – WebSTAR Server Suite V. This Web server is no longer supported by its developer. Users of WebSTAR should consider migrating to the built-in Apache Web server on Mac OS X.

The following procedures describe how to install Lasso Professional Server for each of the Web server configurations above.

To install Lasso Professional Server:

- 1 Double-click the Lasso Pro 8.5 Installer.mpkg application. This launches the Lasso Professional Server window.
- 2 The Lasso Professional Server release notes will be displayed. The release notes contain important late breaking information that might not be covered in the documentation. After reading the release notes which describe what is included with Lasso Professional Server, select Continue.
- 3 The Lasso Professional Server License Agreement will be displayed. After reading and agreeing to the terms, select the Continue and then the Agree button.
- 4 Select the drive you wish to install Lasso Professional Server on. This should be the drive that Mac OS X or Mac OS X Server is installed on. Then select Continue.
- 5 The next steps depends on which of the supported Web servers is being used.
Most users will want to simply select the Install or Upgrade button. This will install the connector for the default version of Apache included with Mac OS X 10.5 or Mac OS X Server 10.5, Mac OS X 10.4, or Mac OS X 10.3.9.
Users with Mac OS X Server 10.4, Mac OS X Server 10.3.9, or with one of the other supported Apache 2 installations or WebSTAR V should select the Customize button. Deselect Lasso Connector for Apache and select the appropriate Web server connector (see above for descriptions). Finally, select the Install or Upgrade button.
- 6 An authentication dialog will prompt for an administrator name and password. Enter the username and password of a Mac OS X user on the machine who has administrative rights. Select Continue.
- 7 Select Quit when the installer has completed. Upon quit the installer automatically restarts the Apache Web server and starts Lasso Service.

After the installers has finished, it will automatically open a Web browser window and attempt to launch the Lasso Initialization page, which is described in the following *Initialization* section.

Initialization

The Initialize page is where the administrator enters the Lasso Professional Server serial number, and sets the site administrator username and password for the first time. To launch the Initialize page, make sure Lasso Service is running and visit <http://localhost/ServerAdmin.LassoApp> in a Web browser. For instructions on how to start and stop Lasso Service, see the *Running Lasso Professional Server* section of this chapter.

Troubleshooting: If in the event that the Lasso Initialization page does not load, make sure that Lasso Service has started properly as described in the *Running Lasso Professional Server* section of this chapter. Otherwise, consult the *Extended Configuration* section to ensure that the installer has configured the Web server correctly (i.e. Apache, WebSTAR).

Initializing Lasso Professional Server

The Welcome to Lasso Professional Server panel displays a message listing the steps required to initialize Lasso. This involves entering the serial number and setting the server administrator username and password.

Figure 1: Server Initialize Page



If no serial number is entered then Lasso will default to Lasso Developer functionality which allows the development of Lasso-based Web sites. Lasso Developer will give 5 IPs to connect and has a per minute limit on the number of connections which it will serve. A Lasso Professional Server serial number is required to deploy Web sites with Lasso.

An evaluation serial number can be used to test the full functionality of Lasso for a limited time prior to purchase. To request an evaluation serial number, select the Request Evaluation Serial Number button. This will take you to the LassoSoft Web site, where an evaluation serial number can be obtained.

To initialize Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number in the Serial Number field. This must be entered exactly as it appears in your LassoSoft invoice or email using the following format:

LP8.5-MAC-123456789

Serial numbers are case-sensitive. Make sure that all capitalization is correct and that all letters and numbers are entered, including the LP8.5 prefix.

- 2 Enter your desired server administrator username in the Administrator Username field. This username will be used by the Lasso server administrator to log in to the administration interface, as discussed in the *Using Server Administration* chapter.

Note: The server administrator username and password in Server Administration are independent of Mac OS X administrator usernames and passwords, and do not have to match.

- 3 Enter your desired server administrator password in the Administrator Password field and re-enter the server administrator password in the Confirm Password field.
- 4 Select Submit Information to continue.

After the initialization information has been successfully submitted, an Initialization Complete panel is displayed stating that Lasso Professional Server has been successfully initialized.

Selecting the Open Server Administration button will open Server Administration, where server preferences can be established. Server Administration also provides access to the Site Administration interface for the default site.

Figure 2: Server Administration



At this point, Lasso Professional Server has been initialized and is ready to use. Lasso Professional Server provides a convenient, Web-based interface for configuring Lasso site settings, instantly managing and editing databases, configuring Lasso Security, monitoring events, and much more. It is within this interface that all Lasso settings and databases are set up and configured.

For a full description of Server Administration and how to use it, see the *Using Server Administration* chapter. For a description of Site Administration and how to use it see the *Using Site Administration* chapter.

The following describes where to go for more information on setting up and testing external data sources, and starting development.

- Setting up existing FileMaker Pro, MySQL, and JDBC data sources for use with Lasso Professional Server is described in the *Setting Up Data Sources* chapter.
- For an overview of setting up Lasso Security for databases, Lasso tags, files, and solutions, see the *Setting Up Security* chapter.
- To learn how to write custom Lasso solutions using Lasso code, see the Lasso 8.5 Language Guide.

Security Note: Lasso Professional Server uses ports 14552 for Lasso Service. Lasso Service cannot be directly accessed by anonymous users, however, it is recommended that system administrators block this port from incoming requests from IP addresses other than the IP address of the Web server using Lasso Professional Server.

Running Lasso Professional Server

Lasso Service is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service is started and stopped using the `lasso8ctl` command or the terminal command files provided in the `/Applications/Lasso Professional 8/Tools` folder.

To start Lasso Service:

- Type the following command into the Mac OS X Terminal application. You will be prompted for your password. This command must be accessed from an administrator account.


```
sudo lasso8ctl start
```
- Alternately, open the Tools folder in the Lasso Professional Server application folder and double click on the `startLassoService.command` file. This will launch a terminal window that prompts you for your password and starts Lasso Service.

Lasso Service can also be run in console mode within the terminal. This allows the administrator to view the Lasso Professional Server status window, which shows all Lasso Professional Server actions that are currently being performed, as well as actions that have been performed since start up. This is useful for monitoring and troubleshooting purposes. Lasso Service should be started this way whenever startup, stability, or performance problems are encountered in Lasso Professional Server.

To start Lasso Service in console mode:

Open the Tools folder in the Lasso Professional Server application folder and double click on the consoleLassoService.command file. This will launch a terminal window that prompts you for your password and starts Lasso Service.

The terminal window contains all of the output of the Lasso Service console. Closing the terminal window will stop Lasso Service. To restart Lasso Service as a background service type Ctrl-C in the terminal to stop Lasso Service. Then type the following command. You will be prompted for your password. This command must be accessed from an administrator account.

```
sudo lasso8ctl restart
```

To stop Lasso Service:

- Type the following command into the Mac OS X Terminal application. You will be prompted for your password. This command must be accessed from an administrator account.

```
sudo lasso8ctl stop
```

- Alternately, open the Tools folder in the Lasso Professional Server application folder and double click on the startLassoService.command file. This will launch a terminal window that prompts you for your password and starts Lasso Service.

To see if Lasso Service is running:

- Type the following command into the Mac OS X Terminal application. You will be prompted for your password. This command must be accessed from an administrator account.

```
sudo lasso8ctl status
```

- Look for a process named Lasso8Service in the Mac OS X Process Viewer utility application.
- Alternately, run the following command in the Mac OS X Terminal to check for Lasso Service in the list of current processes.

```
ps -ax | grep Lasso8Service
```

Terminal Note: See the Mac OS X Tips.pdf file in the Lasso Professional Server application folder Documentation folder for instructions on how to do this.

Extended Configuration

This section discusses the changes the Lasso Professional Server installer makes to the various Web servers it supports, how to configure Lasso Professional Server for a distributed architecture, how to configure WebSTAR V file permissions, and how to install Lasso with J2EE support. This section describes the following:

- *Remote Installation* describes how to install Lasso Professional Server on a remote server.
- *Lasso Connector for J2EE* describes how to manually install and configure Lasso's support for J2EE servers.
- *Lasso Connector for Apache* describes configuring the module that installs into Apache and provides connectivity to Lasso Service.
- *Remote Apache Configuration* describes manually configuring Lasso Service to run on a separate machine from Apache Web server (distributed architecture).
- *Lasso Connector for WebSTAR* describes configuring the module that installs into WebSTAR and provides connectivity to Lasso Service
- *Remote WebSTAR Configuration* describes manually configuring Lasso Service to run on a separate machine from WebSTAR V (distributed architecture).
- *Setting Up Redundant Lasso Professional Server Servers* provides tips for setting up multiple Lasso Professional Server servers in a RAIC.

Important: It is not required that the instructions in this section be followed as part of a default installation or configuration process. The installer will configure a basic installation of Lasso Professional Server properly without any additional modification. This section is intended as an extended configuration reference for experienced administrators.

Remote Installation

For UNIX administrators that desire it, Lasso Professional Server can be installed remotely using ssh. This is useful if you do not have direct access to the Web server where Lasso will be installed, or if the Web server is not set up with a monitor, keyboard, or mouse.

To install Lasso Professional Server on a remote server:

- 1 Copy the Lasso Pro 8.5 Installer.mpkg installer to a location on your remote server. This can be done using any remote file transfer method, such as file sharing, ftp, or scp.
- 2 Use the terminal to ssh into remote server using an administrative user account. An example ssh login command is shown below.

```
ssh administrator@www.example.com
```

- 3 To install Lasso Professional Server, enter the following command. Replace `/path-to-installer/` with the full UNIX folder path to the Lasso Professional Server installer. Enter the administrative user password if prompted.

```
sudo installer -pkg "/path-to-installer/Lasso Pro 8.5 Installer.mpkg" -target /
```

- 4 Once the installer is finished, open your browser and access the remote server default Web page to initialize Lasso Service.

```
http://www.example.com/ServerAdmin.LassoApp
```

Lasso Connector for J2EE Web Server

The Lasso Connector for J2EE provides a means for serving Lasso solutions in a Java 2 Enterprise Edition environment. The connector is a standard Java servlet and can be hosted by any Java HTTP server or servlet container conforming to the Servlet API Specification v2.3 or later.

The connector can be found in the Extensions folder in the Lasso Professional Server application folder. The connector consists of three files:

- Lasso8ConnectorforJ2EE.jar – This Java archive includes the connector itself.
- commons-fileupload.jar – This Java archive includes the upload component for the connector. This file is necessary if [File_Upload] support is desired.
- web.xml – An example XML configuration file.

The installation procedure may vary between different server products, both in terms of location and deployment method (“packed” versus “unpacked”).

Servers that conform to the Servlet API Specification v2.2 (or later) are required to accept a Web Application Archive in a standard format, which is discussed further below.

A web application is defined as a hierarchy of directories and files in a standard layout. Such a hierarchy can be accessed in its “unpacked” form, where each directory and file exists in the filesystem separately, or in a “packed” form known as a Web Archive, or WAR file. The former format is more useful during development, while the latter is used when you distribute your application to be installed.

A special directory exists within the application hierarchy named WEB-INF. This directory contains all things related to the application that aren’t in the document root of the application. No file contained in the WEB-INF directory is served directly to a client.

The contents of the WEB-INF directory are:

- /WEB-INF/classes/ directory for servlet and utility .class files.

- /WEB-INF/lib/ directory for .jar (Java Archive) files which contain servlets, Javabeans, and other utility classes useful to the web application.
- /WEB-INF/web.xml web application deployment descriptor file in XML format

The Lasso Connector archive Lasso8ConnectorforJ2EE.jar and supporting file upload component commons-fileupload.jar must be placed in the /WEB-INF/lib/ folder located in your web application directory.

Additionally, the web application descriptor web.xml must be edited to provide connector configuration parameters. By default, the sample web.xml file is configured to process .lasso and .LassoApp files only. If one wishes a web server to process Lasso files with alternate file extensions other than .lasso or .LassoApp, perform the procedure below.

Java Server Configuration

This section describes setting up the J2EE connector where Lasso Service runs on the same machine as the Java server.

To setup Lasso Connector for J2EE on a remote server:

- 1 Stop Lasso Service using the stopLassoService.command file in the Lasso Professional Server Tools folder.
- 2 Stop the Java Web service if it is running. The instructions for stopping the Java Web service will vary depending on what Java Web service is being used.
- 3 On the machine running Lasso Service, locate the Lasso8ConnectorforJ2EE.jar file and move it into the WEB-INF/lib/ folder on the remote server.
- 4 If your Lasso solution requires HTTP file upload functionality, move optional commons-fileupload.jar file into the WEB-INF/lib/ folder on the remote server.
- 5 If you are adding Lasso support to an existing web application, copy the settings contained within the <web-app> tags in the example web.xml file into the same location in the existing web.xml file. When deploying an entirely new web application, simply move the example deployment descriptor into WEB-INF/ directory on the remote Web server.
- 6 To configure the server to process alternate file extensions with Lasso, insert additional <servlet-mapping> tags in the web.xml file, providing your custom file extension in the url-pattern attribute:

```
<servlet-mapping servlet-name='lasso-connector' url-pattern='*.myExt' />
```

Remote Java Server Configuration

This section describes setting up Lasso Professional Server in a distributed architecture where Lasso Service runs on a separate machine from the Java server.

To setup Lasso Connector for J2EE on a remote server:

- 1 On the machine running Lasso Service, stop Lasso Service using the stopLassoService.command file in the Lasso Professional Server Tools folder.
- 2 On the machine running Lasso Service, stop Java web service if it is running. Since the Lasso Connector for J2EE is being moved to a remote machine, all Lasso pages and LassoApps (including Lasso Administration) should no longer be served from this machine.
- 3 On the machine running Lasso Service, locate the Lasso8ConnectorforJ2EE.jar file and move it into the WEB-INF/lib/ folder on the remote server.
- 4 If your Lasso solution requires HTTP file upload functionality, move optional commons-fileupload.jar file into the WEB-INF/lib/ folder on the remote server.
- 5 If you are adding Lasso support to an existing web application, copy the settings contained within the <web-app> tags in the example web.xml file into the same location in the existing web.xml file. When deploying an entirely new web application, simply move the example deployment descriptor into WEB-INF/ directory on the remote Web server.

- 6 Modify the host attribute of the servlet init-param tag in the web.xml file, supplying the IP address or host name of the computer running the Lasso Professional Server.
- 7 To configure the server to process alternate file extensions with Lasso, insert additional <servlet-mapping> tags in the web.xml file, providing your custom file extension in the url-pattern attribute:

```
<servlet-mapping servlet-name='lasso-connector' url-pattern='*.myExt' />
```

Lasso Connector for Apache

Lasso Connector for Apache is one of the default Web server connectors included with Lasso Professional Server. This section describes what settings Lasso Professional Server adds to the Apache configuration file (httpd.conf), and how to view or manually change the Apache settings.

Apache Configuration

The Lasso Professional Server installer creates a file named lasso8.conf, which is automatically included in the httpd.conf file whenever Apache starts up. This file includes Lasso's default settings and should not be edited.

The following procedure describes viewing and editing the lasso8.conf file, which contains Apache's settings for Lasso.

To view or edit changes made to Apache:

- 1 Start the Mac OS X terminal application, located in Applications/Utilities by default. Expand the terminal window to be as large as possible on the screen for best results.
- 2 Enter the following command to open the lasso8.conf file via the terminal depending on which version of Mac OS X you are using.

Mac OS X 10.5 and Mac OS X 10.5 Server include Apache 2.2 and should use the following command.

```
sudo pico /etc/apache2/other/lasso8.conf
```

Mac OS X 10.4 and Mac OS X 10.3.9 include Apache 1.3 and should use the following command.

```
sudo pico /etc/httpd/users/lasso8.conf
```

Other installations of Apache 2 will be similar, but may store their configuration files in a different location. Consult the documentation of the installation on the server for more information.

Note: This file is locked by default, and one must be logged in as the root user to edit it. See the Mac OS X Tips.pdf file in the Lasso Professional Server Documentation folder for instructions on how to perform these tasks using the terminal.

- 3 Apache can be set up to process alternate file extensions with Lasso Service by adding lines to the lasso8.conf file using the following format: AddHandler lasso8-handler extension. Replace extension with the file extension you wish to be processed by Lasso Service (e.g. .htm, .html, etc.)

Note: The lasso8.conf file contains three lines which define the default settings for Lasso, as shown below. The last two lines tell Apache to process files with the extensions .lasso and .lassoapp with Lasso Service.

Mac OS X 10.5 and Mac OS X 10.5 Server include Apache 2.2 and should use the following configuration directives. Note that additional URLs can be sent to Lasso for processing by adding them using additional location directives.

```
LoadModule lasso8_module /usr/libexec/apache2/Lasso8ConnectorforApache.so
Location ~ "^\.[Ll][Aa][Ss][Ss][Oo]$"
    SetHandler lasso8-handler
</Location>
<Location ~ "^\.[Ll][Aa][Ss][Ss][Oo][Aa][Pp][Pp]$" >
    SetHandler lasso8-handler
</Location>
```

Mac OS X 10.4 and Mac OS X 10.3.9 include Apache 1.3 and should use the following configuration directives. Note that additional URLs can be sent to Lasso for processing by adding them using additional location directives.

```
LoadModule lasso8_module /usr/libexec/httpd/Lasso8ConnectorforApache.so
Location ~ "^\.[Ll][Aa][Ss][Ss][Oo]$"
    SetHandler lasso8-handler
</Location>
<Location ~ "^\.[Ll][Aa][Ss][Ss][Oo][Aa][Pp][Pp]$" >
    SetHandler lasso8-handler
</Location>
```

Other installations of Apache 2 will be similar, but may store their configuration files in a different location. Consult the documentation of the installation on the server for more information.

- 4 Save the `lasso8.conf` file after making any changes.
- 5 Restart Apache Web Server for any changes to take place. This is done by accessing System Preferences/File Sharing in Mac OS X, and then turning Web Sharing off and then on again. Alternately, one can restart Apache Web Server by entering the following command in the terminal:

```
sudo apachectl graceful
```

Remote Apache Configuration

This section describes setting up Lasso Professional Server in a distributed architecture where Lasso Service runs on a separate Mac OS X computer from the Mac OS X Apache Web server. To do this, follow the procedure below.

To set up Lasso Connector for Apache on a remote server:

- 1 On the machine running Lasso Service, stop Lasso Service using the `stopLassoService.command` file.
- 2 On the machine running Lasso Service, stop Apache if it is running. Since the Lasso Connector for Apache is being moved to a remote machine, all Lasso pages and LassoApps (including Site Administration) should no longer be served from this machine.
- 3 On the machine running Lasso Service, locate the `Lasso8ConnectorforApache.so` file and move it into the corresponding `/usr/libexec/httpd` folder or `/usr/libexec/apache2` on the remote server. For instructions how to move hidden system files, see the `Mac OS X Tips.pdf` document.
- 4 On the machine running Lasso Service, locate the `lasso8.conf` file and move it into the corresponding `/etc/httpd/users` folder or `/etc/apache2/other` folder on the remote server. For instructions how to move hidden Unix files, see the `Mac OS X Tips.pdf` document.
- 5 On the machine running Lasso Service, locate the `editLassoApacheconf.command` file in the `/Applications/Lasso Professional 8/Tools` folder, and move it to any location on the remote Web server.
- 6 On the remote server, open the copied `Lasso8Apache.conf` file in the Pico file editor using the following command:

```
sudo pico /etc/lasso/lasso8apache.conf
```

- 7 Next to `Server`, replace `127.0.0.1` with the correct host name or IP address of the machine running Lasso Service.
- 8 Press `Ctrl-X`. This will ask you if you want to save the file.
- 9 Enter `y`. This saves and closes the file.
- 10 On the remote server, restart Apache Web Server. This is done by accessing **System Preferences > File Sharing** in Mac OS X, and then turning Web Sharing off and then on again. Alternately, one can restart Apache Web Server by entering the following command in the terminal:

```
sudo apachectl graceful
```

- 11 On the machine running Lasso Service, start Lasso Service. This can be done using the `startLassoService` command file described in the *Running Lasso Professional Server* section of this chapter.

Note: Only one Web server may connect to the machine running Lasso Service at a time.

Lasso Connector for WebSTAR V

The Lasso Connector for WebSTAR V is represented by a file named `Lasso8ConnectorforWebSTARV.so` that is installed to the WebSTAR V Plug-Ins folder. This file automatically adds all Lasso settings to WebSTAR V.

Some additional configuration may be necessary in order to allow both WebSTAR and Lasso to access files. WebSTAR runs as the user `webstar` and Lasso runs as the user `lasso` in the group `staff`. Once permissions have been set up properly the WebSTAR installer should maintain those permissions on re-installs.

To configure permissions so files can be read by WebSTAR and Lasso:

- Change the `4DWebSTAR`, `4DWebSTAR/WebServer`, and `4DWebSTAR/WebServer/DefaultSite` folders to have world read and execute permissions. This may need to be applied to other sites you have running in the `4DWebSTAR/WebServer` folder, and possibly files contained in those folders.

The following commands can be typed into the Terminal in order to modify the permissions of the folders. Enter the machine's administrator password when prompted.

```
sudo chmod o+rx /Applications/4DWebSTAR
sudo chmod o+rx /Applications/4DWebSTAR/WebServer
sudo chmod o+rx /Applications/4DWebSTAR/WebServer/DefaultSite
```

- Change the group on all files in the `4DWebSTAR` folder to `staff`. The `lasso` user is part of the `staff` group and making this change would allow Lasso to access the files. Unfortunately, users not in the `staff` group would not be able to get in. In Mac OS X 10.2 all users belonged to the `staff` group, but in 10.3.9, no users are added to the `staff` group when they are created.

The following command can be typed into the Terminal in order to modify the group owner of all files within the `4DWebSTAR` folder. Enter the machine's administrator password when prompted.

```
sudo chgrp -R staff /Applications/4DWebSTAR
```

By default, Lasso Connector for WebSTAR V is configured to process `.lasso` and `.LassoApp` files only. These extensions are built into Lasso Connector for WebSTAR V and cannot be changed. If one wishes WebSTAR to process Lasso files with alternate file extensions other than `.lasso` or `.LassoApp`, perform the procedure below.

To configure WebSTAR to process alternate file extensions with Lasso:

- 1 Launch the 4D WebSTAR Admin Client application and connect to the 4D WebSTAR Admin Server.
- 2 To make global changes to all Web sites, select the Admin option from the list of available virtual hosts. Otherwise, select `DefaultSite` (or another virtual host).
- 3 Select the Select button.
- 3 Select Suffix Mappings option from the Web Server section on the left panel of the Administration window.
- 4 In the Editable Suffix Mappings panel, add as many suffix mappings as necessary by specifying the filename extension, MIME type, and choosing the Lasso action from the popup menu at the bottom of the window.
- 5 When finished, select the Save button to save the changes.

Remote WebSTAR V Configuration

This section describes setting up Lasso Professional Server in a distributed architecture where Lasso Service runs on a separate Mac OS X computer from the WebSTAR V server.

To setup Lasso Connector for WebSTAR V on a remote server:

- 1 On the machine running Lasso Service, stop Lasso Service using the `stopLassoService.command` file in the Lasso Professional Server/Tools folder.
- 2 On the machine running Lasso Service, stop WebSTAR V if it is running. Since the Lasso Connector for WebSTAR V is being moved to a remote machine, all Lasso pages and LassoApps (including Site Administration) should no longer be served from this machine.
- 3 On the machine running Lasso Service, locate the `Lasso8ConnectorforWebSTAR.bundle` file and move it into the `4DWebSTAR/WebServer/Plug-Ins` folder on the remote server.
- 4 On the machine running Lasso Service, locate the hidden `Lasso8WebSTAR.conf` file in the `/etc/lasso` folder, and copy it to the `/etc/lasso` folder on the remote Web server.
- 5 On the remote server, open the copied `Lasso8WebSTAR.conf` file in the Pico file editor using the following command:

```
sudo pico /etc/httpd/users/Lasso8WebSTAR.conf
```
- 6 Next to `Server`, replace `127.0.0.1` with the correct host name or IP address of the machine running Lasso Service.
- 7 Press `Ctrl-X`. This will ask you if you want to save the file.
- 8 Enter `y`. This saves and closes the file.
- 9 On the remote Web server, launch WebSTAR V.
- 10 On the machine running Lasso Service, start Lasso Service using the `startLassoService.command` file or `consoleLassoService.command` file, as described in the *Running Lasso Professional Server* section of this chapter.

Note: In Lasso Professional Server, only one Web server may connect to the machine running Lasso Service at a time.

Setting Up Redundant Lasso Professional Server Servers

For load balancing purposes, multiple Web servers running Lasso Professional Server may be set up in a RAIC (Redundant Array of Independent Components) in the same manner as other Web servers. There are a variety of ways to do this depending on the networking hardware and software available, and instructions for setting up a RAIC are beyond the scope of this guide.

However, there is one setting in Lasso Professional Server that must be adjusted in order for Lasso sessions to persist across a RAIC, and that is to point each instance of Lasso Professional Server to a single MySQL sessions server. For instructions on how to do this, please see the *Setting Site Preferences* chapter.

Installation Contents

This section contains a list of all files and folders that are installed during Lasso Professional Server installation.

Lasso User Note: The Lasso Professional Server installer creates a hidden `Lasso` user in Mac OS X that is used to read, write, and execute files related to Lasso Professional Server. This user is installed automatically, and no configuration for this user is required. Do not attempt to configure or change the settings for this user as it will affect the functionality of Lasso Professional Server. For more information about the `Lasso` user, see the `Mac OS X Tips.pdf` file.

Application Files

Lasso Professional Server installs the following files and folders in the `Applications/Lasso Professional Server` folder on the Mac OS X hard drive.

- **Chart FX folder** – Contains additional documentation about the Chart FX module and the Chart FX for Java Designer which can be used to create Chart FX template files.
- **Documentation folder** – Contains all Lasso Professional Server documentation in PDF format. The following files and folders are contained in this folder. It is highly recommended that the documentation be read in this order.
 - 1-ReadMeFirst folder – Contains Release Notes.txt, License Agreement.txt, and Mac OS X Tips.pdf. Read this documentation in this folder first before proceeding to configure Lasso Professional Server.
 - 2-SetupGuide folder – Contains Lasso 8.5 Setup Guide.pdf.
 - 3-LanguageGuide folder – Contains Lasso 8.5 Language Guide.pdf, and the LassoApps, LCP, LCAPI, and LJAPI folders.
- **Extensions folder** – Contains additional optional components.
 - J2EE folder – Contains the Lasso Connector for J2EE which is required for Java Enterprise Server support.
 - Apache 2 folder – Contains the Lasso Connector for Apache 2 which is required for manual installation of support for Apache 2.
 - Apache 2.2 folder – Contains the Lasso Connector for Apache 2.2 which is required for manual installation of support for Apache 2.2.
 - OS Process folder – Contains the [OS_Process] module which allows access to the shell or other processes running natively on each platform.
- **ImageMagick folder** – Contains an installation of ImageMagick, which is used to power the [Image] tags in Lasso 8. For more information, see the *Images and Multimedia* chapter in the Lasso 8.5 Language Guide.
- **JavaLibraries folder** – Any extra Java libraries (usually .jar files) can be placed in this folder. They will be loaded when Lasso starts Java, but no JDBC drivers or tag modules will be loaded from them.
 - iText.jar – Required library for the [PDF_...] tags.
- **JDBCDrivers folder** – Drivers for JDBC data sources are placed here (empty by default). For instructions on how to use this folder while configuring JDBC data sources, see the *Setting Up Data Sources* chapter.
- **LassoAdmin folder** – Contains files and folders used for administering Lasso Professional Server.
 - BuildLassoApps folder – Is a folder used to build LassoApps. For more information, see the *Site Administration Utilities* chapter.
 - Lasso8Service.sh – This is the “watchdog” script for Lasso Professional Server. This file continuously queries Lasso Service to make sure that it is running when it is supposed to be, and automatically restarts Lasso Service if it inadvertently quits. When it does this, a log entry is added to the standard system log file (traditionally /var/log/system.log) in Mac OS X. If the number of restarts exceeds 20 per hour, the script stops attempting to start Lasso Service until the problem is fixed manually and the machine is rebooted.

Note: While Lasso Service is running, the Lasso8Service.sh file will temporarily place a file named LassoService8.pid in the /tmp/ folder. This file is a helper file for Lasso8Service.sh.
- **LassoApps folder** – Contains LassoApps for on-demand loading. When a LassoApp file is requested this folder is checked first and the LassoApp is loaded from here if it exists. This allows LassoApps to be served without placing them in the Web server root.
- **LassoLibraries folder** – Contains LassoApps and .Lasso pages for on-demand loading. Each file should have the name of the namespace it defines. When Lasso tries to execute an unknown tag it checks here to see if it can load the tag before continuing.
- **LassoModules folder** – Contains all default Lasso modules included with Lasso Professional Server as listed below. All data source connectors (including third-party connectors) and tag modules should go here.
 - chartfx.lasso.jar – Implements Chart FX support.
 - FMPConnector.dylib – Data source connector module for FileMaker Pro data sources.
 - FMSAConnector.dylib – Data source connector module for FileMaker Server data sources.
 - ImageTag.dylib – Implements support for the image tags..

LDAPConnector.dylib – Implements support for querying LDAP data sources.

LJAPI.dylib – Implements core support for all LJAPI modules and includes the data source connector for JDBC data sources.

LJAPI.jar – Implements core support for all LJAPI modules.

MySQLConnector.dylib – Data source connector module for MySQL data sources.

NSLookup.class – Java class file for [NSLookup] tag.

OpenBaseConnector.dylib – Data source connector module for OpenBase data sources.

PDF.jar – Required library for the [PDF_...] tags.

SpotlightConnector.dylib – Implements support for querying the spotlight disk indexing in Mac OS X 10.4/10.5.

SQLConnector.dylib – Implements support for SQL-based data sources including Oracle, PostgreSQL, and Microsoft SQL Server.

Note: There is no separate module for the Lasso Connector for SQLite. This is built into the core Lasso Service code.

- **Lasso8Service** – Lasso8Service is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service is the Lasso Web Data Engine.
- **LassoSites folder** – Contains the site level folders for each site that has been created. By default contains a default-1 folder for the default site. This folder contains most of the same folders as the Lasso Professional Server application folder (called the master level). In general, the site level is checked before the master level when Lasso needs to find a resource.
- **LassoStartup folder** – Contains the Startup.LassoApp file and sites folder, which are used by Lasso Service during startup. All Lasso pages and LassoApps which need to be launched during startup should be placed here. For information on how to use the LassoStartup folder, see the *Lasso pages* chapter of the Lasso 8.5 Language Guide. For additional discussion of the LassoStartup folder, see the Lasso 8.5 Language Guide.
- **SQLiteDBs folder** – Contains SQLite databases used to store Lasso Professional Server settings and preferences.
- **Tools folder** – Contains Mac OS X terminal command files for performing various tasks with Lasso Professional Server. These command files allow users to perform tasks using the terminal by simply double-clicking on the command files.

consoleLassoService.command – Starts Lasso Service as an application in the terminal. Useful for viewing the startup log and troubleshooting. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

startLassoService.command – Starts Lasso Service via the terminal. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

stopLassoService.command – Stops Lasso Service via the terminal. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

StartupItems

The Library/StartupItems folder contains files that are used to auto-start Lasso Service at boot time.

- **LassoPro8** – Contains the LassoPro8 and StartupParameters.plist files which are used to auto-start Lasso Service at boot time.

Framework

The Library/Frameworks folder contains the Lasso.Framework.

- **Lasso.Framework** – Contains code that Lasso Service needs to execute properly. Most of Lasso's functionality is implemented within this framework.

Web Server Root

Lasso Professional Server does not install any files in the Web server root. All of Lasso's administration tools are loaded on-demand from the LassoApps folder so do not need to be located in this folder.

System Files

Lasso Professional Server installs the following files in various system locations on the Mac OS X hard drive.

Note: These files are hidden unless viewed in the terminal application, or via a file editor capable of viewing hidden system files.

- **Lasso8ConnectorforApache.so** – This file is the Lasso Web server connector for Apache 1.3, which is installed in the in `/user/libexec/httpd` folder for default installations of Mac OS X 10.4 or Mac OS X 10.3.9.
- **Lasso8ConnectorforApache2.2.so** – This file is the Lasso Web server connector for Apache 2.2, which is installed in the in `/user/libexec/apache2` folder for default installations of Mac OS X 10.5 or Mac OS X Server 10.5.
- **Lasso8ConnectorforApache2.so** – This file is the Lasso Web server connector for Apache 2, which is installed in different locations if one of the Apache 2 options is selected during installation.
- **Lasso8ConnectorforWebSTAR.bundle** – This file is the Lasso Web server connector for WebSTAR V, which is installed in the WebSTAR Plug-Ins folder if the option is selected during installation.
- **lasso8.conf** – Lasso Professional Server adds an Apache configuration file to the `/etc/httpd/users` folder. This file is dynamically included in the main Apache configuration file (`httpd.conf`).
- **lasso8apache.conf** – The Web server configuration file for the Apache connectors, located in the `/etc/lasso` folder.
- **lasso8webstar.conf** – The Web server configuration file for Lasso Connector for WebSTAR, located in the `/etc/lasso` folder.
- **ICU Libraries** – These required files are installed to the `/private/tmp/lassoicu/` folder. These are necessary for Unicode support.

Uninstalling Lasso Professional Server

Lasso Professional Server should be uninstalled using the uninstaller application provided as part of the Lasso Professional Server package. Following the procedure below will ensure that all custom settings and databases will be preserved.

To uninstall Lasso Professional Server:

- 1 Perform an export of your current Server Administration settings using the *Setup > Site > Import/Export* section of Server Administration. Save the exported file somewhere on your hard drive.
- 1 Double-click the Lasso Pro 8 Uninstaller.mpkg application. This launches the Lasso Professional Server window. An authentication dialog will prompt for an administrator name and password.
- 2 Enter the username and password of a Mac OS X user on the machine who has administrative rights.
- 3 Select Continue. This will display special information about what the unintaller will do.
- 4 Select Continue, then select the drive you wish to uninstall Lasso Professional Server from.
- 7 Select Continue button, then select Upgrade. This will remove Lasso Professional Server from your system.
- 8 Select Quit when the uninstaller has completed.

Files and Folders Not Removed

The uninstaller program will not remove any non-default files created after the time of install, or any default Lasso files modified since the time of install. These files include the SQLite databases, Lasso log files, Lasso setup files, and custom Lasso pages and LassoApps. These files and folders may be backed up for preservation, or may be deleted to completely remove Lasso Professional Server and all settings from the system.

- **Application folder** – The Applications/Lasso Professional Server folder is left on the hard drive, which contains the following folders:
 - JavaLibraries – Contains any Java libraries installed by the administrator. This folder will be removed if no extra Java libraries were installed.
 - JDBCDrivers – Contains any JDBC drivers installed by the administrator. This folder will be removed if no JDBC drivers were installed.
 - LassoApps – Contains LassoApps for on-demand loading. This folder will be removed if no extra LassoApps were installed.
 - LassoLibraries – Contains tag libraries for on-demand loading. This folder will be removed if no extra tag libraries were installed.
 - LassoModules – Contains data sources connectors, Java libraries, and any third-party Lasso modules.
 - LassoSites – Contains the site level folders for each defined site. The site level folders have the same structure as the master level folders described here. Each site might have its own JavaLibraries, JDBCDrivers, LassoApps, LassoLibraries, LassoModules, and SQLiteDBs.
 - SQLiteDBs – Contains all default SQLite databases, which contain serial numbers, passwords, database settings, logs, and permissions from the previous setup of Lasso Professional Server.
- **Web Server Root folder** – The Web server root folder retains all Lasso pages created or modified since the time of installation. Customized Lasso pages, Web pages, and scripts will not be deleted during uninstallation.

Reinstallation Note: Any files remaining on the system from a previous installation of Lasso Professional Server will not be overwritten by installing a newer version. Therefore, all previous settings will be retained if a newer version of Lasso Professional Server is installed over the files not removed during the uninstallation of the previous version.

4

Chapter 4 Configuring on Windows

This chapter contains the installation and configuration instructions for Lasso Professional Server on Microsoft Windows. It also contains information on Lasso Professional Server components and how they can be manually installed or modified.

- *System Requirements* lists the minimum system requirements for Lasso Professional Server.
- *Web Browser Requirements* lists the Web browser requirements for administering Lasso Professional Server.
- *Installation Instructions* includes step-by-step instructions for installing and initializing Lasso Professional Server, and establishing a basic setup.
- *Running Lasso Professional Server* describes how to start and stop Lasso Service, and how to check if it is running properly.
- *Extended Configuration* describes how to manually configure IIS for Lasso Professional Server, and how to set up a Lasso Professional Server distributed architecture.
- *Installation Contents* lists every file installed with Lasso Professional Server.
- *Uninstalling Lasso Professional Server* includes step-by-step instructions for removing Lasso Professional Server from your system.

System Requirements

Lasso Professional Server will run on systems which meet the following minimum requirements. Although Lasso Professional Server may run on machines which do not meet these requirements, those installations will not be supported.

- 300 MHz or higher Pentium-compatible CPU.
- 512 MB of RAM. More recommended. (Requires an additional 64 MB of available RAM for each additional site. See the *Setting Up Sites* chapter for more information.)
- Windows XP Professional, Windows 2000 family (Professional, Server and Advanced Server editions), or Windows Server 2003 family (Web, Standard, Enterprise and Datacenter editions).
- Microsoft Internet Information Services (IIS) 5 or 6 (included with the versions of Windows listed above), or Apache 2.
- Java Virtual Machine compatible with Sun's JRE 1.4 (Java 2). This is required for the use of LJAPI modules, JDBC data sources, and all Lasso tags which are dependent upon Java. For details on which Lasso tags are dependent upon Java, see the Lasso 8 Reference. For more information on LJAPI, see the Lasso 8.5 Language Guide.

Note: ImageMagick is automatically installed with Lasso Professional 8.5.2 and higher.

Web Browser Requirements

This section describes the basic Web browser requirements needed to successfully set up and administer Lasso Professional Server.

- Microsoft Internet Explorer 5.0 or higher, FireFox, or compatible browser.
- Javascript enabled.
- Cookies enabled.
- Cascading Style Sheets enabled.

If these criteria are not met, then a browser check dialog box will be shown when trying to access Site Administration. You may proceed into the interface without meeting all these criteria, however some elements may not work properly. It is highly recommended that a browser that meets the listed requirements be used.

Installation Instructions

This section discusses installation procedures for Lasso Professional Server, and initializing Server Administration for the first run. This section will introduce the following:

- *Installing Java* describes installing the Java components necessary for Lasso Professional Server Java-based features to function properly.
- *Installing Lasso Professional Server* describes how to install Lasso Professional Server using the auto-installation program.
- *Initialization* describes the initialization of Lasso Professional Server for running the first time.

Important: Instructions for upgrading from Lasso Professional 5, 6, or 7 are included in the *Upgrading* chapter in this section.

Installing Java

This section discusses installing Sun's Java Runtime Environment (JRE) 6. This installation is required to successfully use the Java-based features available in Lasso Professional Server. These include LJAPI modules, JDBC data source connectivity, and Lasso tags which are dependent upon Java. For details on which Lasso tags are dependent upon Java, see the Lasso 8.5 Reference. It is unnecessary to install Java if these features will not be used.

To install JRE 6:

- 1 Download Sun's JRE installer at the following address:

<http://java.sun.com/javase/downloads/index.jsp>

- 2 Double-click the JRE installer application, and then follow the instructions to install the JRE.
- 3 Restart your system for the changes to take effect.

The Java components needed to run the Java-based features in Lasso Professional Server are now installed. The Java-based features of Lasso Professional Server can be set up, enabled, and tested within Site Administration. See the *Setting Site Preferences* chapter *Java* section for more information.

Installing Lasso Professional Server

Use the Lasso Professional Server Installer to ensure that all of the Lasso Professional Server files are installed in the proper location. The following procedure assumes IIS will be used as the Web server.

Upgrading Note: A previous installation of Lasso Professional 8 or 8.5 must be removed prior to installing the new version. Use the Remove option in the installer to remove the previous installation then re-run the installer to install the new version. No databases or settings files will be removed by this procedure.

To install Lasso Professional Server for IIS:

- 1 From the installation CD or the electronic download, double-click the Lasso Pro 8.5 Installer.msi application. This launches the Lasso Professional Server Setup window.
- 2 You must agree to the license agreement before installing Lasso. The release notes screen includes some late-breaking information that may not be contained in the documentation.
- 3 Select the drive and destination folder to which Lasso Professional Server will be installed. This is C:\Program Files\OmniPilot Software\Lasso Professional 8\ by default. To ensure all features in Lasso Professional Server to work properly, it is strongly recommended that the names of these folders not be changed. Select Next.
- 4 A message will be displayed stating that Lasso Professional Server is ready to be installed. Select the Back button to check or modify any of the previous steps. Select Next to install Lasso Professional Server.
- 5 Select Finish when the installer has completed.

Lasso Service will automatically start after the installer has finished and a Web browser window will automatically open to the Lasso Initialization page, which is described in the next section.

Windows Server 2003 Configuration

Windows Server 2003 requires additional configuration of IIS 6. Lasso must be explicitly allowed in the Web Service Extensions configuration in order for IIS to serve .lasso or .lassoapp files.

- 1 Open the Internet Information Services (IIS) Manager.
- 2 Expand the local computer item and select Web Service Extensions.
- 3 Select Add a new Web service extension... from the tasks list or the Action menu.
- 4 Type Lasso Professional for the Extension Name. Click the Add... button and locate the file at the path below; Check the Set extension status to Allowed checkbox. Click OK to continue.
C:\WINDOWS\system32\Lasso8ConnectorforIIS.dll
- 5 A new item Lasso Professional with status Allowed is created.

Additional Configuration

These additional steps are necessary to finish configuring IIS.

- 1 Get Properties on the WebSites folder in the Internet Information Services (IIS) Manager. Under the ISAPI Filters, make sure there is a LassoFilter and its priority is High.
- 2 On the Home Directory tab under Configuration, make sure that both .lasso and .LassoApp are listed. Make sure that both Extensions point to the C:\WINDOWS\system32\Lasso8ConnectorforIIS.dll. For the .LassoApp extension, make sure that the Check that File Exists box is **not** checked.
- 3 Repeat the instructions in step 2 for the Home Directory of the Default Site as well.

Note: Setting the option All Unknown ISAPI Extensions to Allow will also allow Lasso processed Web pages to be served by IIS, but is not recommended since it essentially defeats the Web Service Extensions security feature offered by IIS 6.

Installing Lasso Professional Server With Apache 2

Use the Lasso Professional Server Installer to ensure that all of the Lasso Professional Server files are installed in the proper location. The following procedure includes the additional steps necessary to install Lasso with Apache 2 support.

Note: If you have already installed Lasso 8 with IIS support then you may need to run the installer again and Modify your installation to add the Lasso Connector for Apache 2. Then skip down to step 7 to complete the installation.

To install Lasso Professional Server for Apache 2:

- 1 From the installation CD or the electronic download, double-click the Lasso Pro 8.5 Installer.msi application. This launches the Lasso Professional Server Setup window.
- 2 You must agree to the license agreement before installing Lasso. The release notes screen includes some late-breaking information that may not be contained in the documentation.
- 3 Select the drive and destination folder to which Lasso Professional Server will be installed. This is C:\Program Files\OmniPilot Software\Lasso Professional 8\ by default. To ensure all features in Lasso Professional Server to work properly, it is strongly recommended that the names of these folders not be changed. Select Next.
- 4 Select the Custom... button to change the installer options. Disable the option to install the Lasso Connector for IIS and enable the option for Lasso Connector for Apache 2.
- 5 A message will be displayed stating that Lasso Professional Server is ready to be installed. Select the Back button to check or modify any of the previous steps. Select Next to install Lasso Professional Server.
- 6 Select Finish when the installer has completed.
- 7 Open the Lasso Professional 8 application folder (located in C:\Program Files\OmniPilot Software\Lasso Professional 8\ by default). Open the Extensions folder and the Lasso Connector for Apache 2 folder.
- 8 The file Lasso Connector for Apache 2.dll must be placed in your Apache modules folder which is usually at C:\Program Files\Apache Group\Apache2\modules\.
- 9 The contents of the lasso8.conf file must be merged into your Apache 2 httpd.conf file which is usually in C:\Program Files\Apache Group\Apache2\conf.
- 10 Restart the Apache 2 service using the Services control panel.
- 11 Access the following URL in order to initialize Lasso 8.

<http://localhost/ServerAdmin.LassoApp>

Initialization

The Initialize page is launched in a Web browser after Lasso Professional Server Installer has completed. The Initialize page is where the server administrator enters the Lasso Professional Server serial number, and sets the site administrator username and password for the first time. If this page does not launch automatically, then make sure Lasso Service is running and visit <http://127.0.0.1/ServerAdmin.LassoApp> in a Web browser. For instructions on how to start and stop Lasso Service, see the *Running Lasso Professional Server* section of this chapter.

Troubleshooting: If in the event that the Lasso Initialization page does not load, make sure that Lasso Service and SQLite have started properly as described in the *Running Lasso Professional Server* section of this chapter. Otherwise, make sure that the installer was able to successfully configure IIS as described in the *Extended Configuration > Lasso Connector for IIS* section of this chapter. Also, make sure that the IIS Admin Service and World Wide Web Publishing Service have restarted properly.

Initializing Lasso Professional Server

The Welcome to Lasso Professional Server panel displays a message listing the steps required to initialize Lasso. This involves entering the serial number, setting the server administrator username and password, and establishing the SMTP preferences.

Figure 1: Initialize Page



If no serial number is entered then Lasso will default to Lasso Developer functionality which allows the development of Lasso-based Web sites. Lasso Developer will give five IPs to connect and has a per minute limit on the number of connections which it will serve. A Lasso Professional Server serial number is required to deploy Web sites with Lasso.

An evaluation serial number can be used to test the full functionality of Lasso for a limited time prior to purchase. To request an evaluation serial number, select the Request Evaluation Serial Number button. This will take one to the LassoSoft Web site, where an evaluation serial number can be obtained.

Once a serial number is obtained, Lasso Professional Server can be initialized by following the procedure below.

To initialize Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number in the Serial Number field. This must be entered exactly as it appears in your LassoSoft invoice or email using the following format:

LP8.5-WIN-123456789

Serial numbers are case-sensitive. Make sure that all capitalization is correct and that all letters and numbers are entered, including the LP8.5 prefix.

- 2 Enter your desired server administrator username in the Administrator Username field. This username will be used by the Lasso server administrator to log in to the administration interface, as discussed in the *Using Server Administration* chapter.

Note: The server administrator username and password in Server Administration are independent of Mac OS X administrator usernames and passwords, and do not have to match.

- 3 Enter your desired server administrator password in the Administrator Password field and re-enter the server administrator password in the Confirm Password field.
- 4 Select Submit Information to continue.

After the initialization information has been successfully submitted, an Initialization Complete panel is displayed stating that Lasso Professional Server has been successfully initialized.

Selecting the Open Server Administration button will open Server Administration, where server preferences can be established. Server Administration also provides access to the Site Administration interface for the default site.

Figure 2: Server Administration



At this point, Lasso Professional Server has been initialized and is ready to be used. Lasso Professional Server provides a convenient, Web-based interface for configuring Lasso site settings, instantly managing and editing databases, configuring Lasso Security, monitoring events, and much more. It is within this interface that all Lasso settings and databases are set up and configured.

For a full description of Server Administration and how to use it, see the *Using Server Administration* chapter.

The following describes where to go for more information on setting up and testing external data sources, and starting development.

- Setting up existing FileMaker Pro, MySQL, and JDBC data sources for use with Lasso Professional Server is described in the *Setting Up Data Sources* chapter.
- For an overview of setting up Lasso Security for databases, Lasso tags, files, and solutions, see the *Setting Up Security* chapter.
- To learn how to write custom Lasso solutions using Lasso code, see the Lasso 8.5 Language Guide.

Security Note: Lasso Professional Server uses port 14552 for Lasso Service. Lasso Service cannot be accessed by anonymous users, however, to protect Lasso Service it is recommended that system administrators block these ports from incoming requests from IP addresses other than the IP address of the Web server using Lasso Professional Server.

Enabling Auto Recovery

Windows can be set to automatically restart Lasso Service if it crashes. This is done by configuring the options in the Services control panel.

To configure auto recovery of Lasso Service:

- 1 From the Windows Task Bar, select *Start > Settings > Control Panel > Administrative Tools > Services*.
- 2 Right click on Lasso Professional Server 8, then select Properties (or double-click on the Lasso Service line).
- 3 Select the Recovery tab.
- 4 Set the action on each failure to Restart the Service.

Running Lasso Professional Server

Lasso Service is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service can be started and stopped within the Services menu in Windows.

To start or stop Lasso Service:

- 1 From the Windows Task Bar, select **Start > Settings > Control Panel > Administrative Tools > Services**.
- 2 Right click on Lasso Professional Server 8, then select Start to start Lasso Service, or Stop to stop Lasso Service.

In Windows, one also has the ability to start Lasso Service in console mode. This allows one to view the Lasso Professional Server status window, which shows all Lasso Professional Server actions that are currently being performed, as well as actions that have been performed since start up. This is useful for monitoring and troubleshooting purposes.

To start Lasso Service in console mode:

- 1 Stop Lasso Service if it is running.
- 2 Browse to the Program Files\OmniPilot Software\Lasso Professional Server folder on your hard drive.
- 3 Double-click on LassoService.exe.

Extended Configuration

This section discusses the changes the Lasso Professional Server installer makes to IIS, and how to configure Lasso Professional Server for a distributed architecture with IIS. This section describes the following:

- *Lasso Connector for J2EE* describes how to manually install and configure Lasso's support for J2EE servers.
- *Lasso Connector for IIS* describes configuring the module that installs into Microsoft IIS and provides connectivity to Lasso Service.
- *Remote IIS Configuration* describes manually configuring Lasso Service to run on a separate machine from the Windows Web server with IIS (distributed architecture).
- *Setting Up Redundant Lasso Professional Server Servers* provides tips for setting up multiple Lasso Professional Server servers in a RAIC.

Important: It is not required that the instructions in this section be followed as part of a default installation or configuration process. The installer will configure a basic installation of Lasso Professional Server properly without any additional modification. This section is intended as an extended configuration reference for experienced administrators.

Lasso Connector for J2EE Web Server

The Lasso Connector for J2EE provides a means for serving Lasso solutions in a Java 2 Enterprise Edition environment. The connector is a standard Java servlet and can be hosted by any Java HTTP server or servlet container conforming to the Servlet API Specification v2.3 or later.

The connector can be found in the Extensions folder in the Lasso Professional Server application folder. The connector consists of three files:

- Lasso8ConnectorforJ2EE.jar – This Java archive includes the connector itself.
- commons-fileupload.jar – This Java archive includes the upload component for the connector. This file is necessary if [File_Upload] support is desired.
- web.xml – An example XML configuration file.

Note: If the Extensions folder was not installed then run the installer again and **Modify** the installation to add the Lasso Connector for J2EE.

The installation procedure may vary between different server products, both in terms of location and deployment method ("packed" versus "unpacked").

Servers that conform to the Servlet API Specification v2.2 (or later) are required to accept a Web Application Archive in a standard format, which is discussed further below.

A web application is defined as a hierarchy of directories and files in a standard layout. Such a hierarchy can be accessed in its “unpacked” form, where each directory and file exists in the filesystem separately, or in a “packed” form known as a Web Archive, or WAR file. The former format is more useful during development, while the latter is used when you distribute your application to be installed.

A special directory exists within the application hierarchy named WEB-INF. This directory contains all things related to the application that aren’t in the document root of the application. No file contained in the WEB-INF directory is served directly to a client.

The contents of the WEB-INF directory are:

- /WEB-INF/classes/ directory for servlet and utility .class files.
- /WEB-INF/lib/ directory for .jar (Java Archive) files which contain servlets, Javabeans, and other utility classes useful to the web application.
- /WEB-INF/web.xml web application deployment descriptor file in XML format

The Lasso Connector archive Lasso8ConnectorforJ2EE.jar and supporting file upload component commons-fileupload.jar must be placed in the /WEB-INF/lib/ folder located in your web application directory.

Additionally, the web application descriptor web.xml must be edited to provide connector configuration parameters. By default, the sample web.xml file is configured to process .lasso and .LassoApp files only. If one wishes a web server to process Lasso files with alternate file extensions other than .lasso or .LassoApp, perform the procedure below.

Java Server Configuration

This section describes setting up the J2EE connector where Lasso Service runs on the same machine as the Java server.

To setup Lasso Connector for J2EE on a remote server:

- 1 Stop Lasso Professional Server 8 using the Services control panel.
- 2 Stop the Java Web service if it is running. The instructions for stopping the Java Web service will vary depending on what Java Web service is being used.
- 3 On the machine running Lasso Service, locate the Lasso8ConnectorforJ2EE.jar file and move it into the WEB-INF/lib/ folder on the remote server.
- 4 If your Lasso solution requires HTTP file upload functionality, move optional commons-fileupload.jar file into the WEB-INF/lib/ folder on the remote server.
- 5 If you are adding Lasso support to an existing web application, copy the settings contained within the <web-app> tags in the example web.xml file into the same location in the existing web.xml file. When deploying an entirely new web application, simply move the example deployment descriptor into WEB-INF/ directory on the remote Web server.
- 6 To configure the server to process alternate file extensions with Lasso, insert additional <servlet-mapping> tags in the web.xml file, providing your custom file extension in the url-pattern attribute:

```
<servlet-mapping servlet-name='lasso-connector' url-pattern='*.myExt' />
```

Remote Java Server Configuration

This section describes setting up Lasso Professional Server in a distributed architecture where Lasso Service runs on a separate machine from the Java server.

To setup Lasso Connector for J2EE on a remote server:

- 1 On the machine running Lasso Service, stop Lasso Professional Server 8 using the Services control panel.
- 2 On the machine running Lasso Service, stop Java web service if it is running. Since the Lasso Connector for J2EE is being moved to a remote machine, all Lasso pages and LassoApps (including Lasso Administration) should no longer be served from this machine.

- 3 On the machine running Lasso Service, locate the `Lasso8ConnectorforJ2EE.jar` file and move it into the `WEB-INF/lib/` folder on the remote server.
- 4 If your Lasso solution requires HTTP file upload functionality, move optional `commons-fileupload.jar` file into the `WEB-INF/lib/` folder on the remote server.
- 5 If you are adding Lasso support to an existing web application, copy the settings contained within the `<web-app>` tags in the example `web.xml` file into the same location in the existing `web.xml` file. When deploying an entirely new web application, simply move the example deployment descriptor into `WEB-INF/` directory on the remote Web server.
- 6 Modify the host attribute of the servlet `init-param` tag in the `web.xml` file, supplying the IP address or host name of the computer running the Lasso Professional Server.
- 7 To configure the server to process alternate file extensions with Lasso, insert additional `<servlet-mapping>` tags in the `web.xml` file, providing your custom file extension in the `url-pattern` attribute:

```
<servlet-mapping servlet-name='lasso-connector' url-pattern='*.myExt' />
```

Lasso Connector for IIS

Lasso Connector for IIS is one of the default Web server connectors included with Lasso Professional Server. This section describes what settings Lasso Professional Server adds to Microsoft IIS, and how to manually perform these setting changes. Lasso Professional Server adds the following settings to Microsoft IIS:

- Adds `LassoFilter` to ISAPI filters for Web server.
- Configures the `.Lasso` and `.LassoApp` extensions in WWW Service Application Configuration.

All files required to use Lasso Connector for IIS are located in the `Windows\System32` and `InetPub\wwwroot` folders, as described earlier in this chapter. Do not remove any of these files as doing so will cause Lasso Connector for IIS not to work. Always check the Lasso Professional Server release notes for the most current configuration information.

Certification Note: Lasso Professional Server is certified to work only with default Windows installations of IIS. Altered versions of IIS may work with Lasso Professional Server, but are not officially supported.

Adding the ISAPI Filter

The `Lasso8ConnectorforIIS.dll` file is the ISAPI filter that allows Microsoft IIS to recognize Lasso actions. Follow the procedure below to add or verify the ISAPI filter in the Internet Services Manager, located in *Start > Settings > Control Panel > Administrative Tools* from the Windows Task Bar.

- 1 Using the Internet Services Manager, right-click on the local computer name and select Properties in Windows 2000, or right-click on the Web Sites folder and select Properties in Windows XP.
- 2 Select the Edit button to edit the WWW Services Master properties.
- 3 Select the ISAPI Filters tab.
- 4 If `LassoFilter` does not appear in the list, click the Add button.
- 5 Enter `LassoFilter` in the Filter Name field.
- 6 Select the Browse button and browse to the `Windows\System32` directory, and select `Lasso8ConnectorforIIS.dll`.
- 7 Select the OK button on all open dialogs to apply the changes.
- 8 Right-click on the computer name in the Internet Services Manager and select Restart IIS.

Configuring the .Lasso and .LassoApp Extensions

The `.lasso` and `.LassoApp` extensions must be added to the Microsoft IIS Application Configuration screen in order to process Lasso pages.

- 1 Using the Internet Services Manager, right-click on the local computer name and select Properties from the pop-up menu.
- 2 Select the Edit button to edit the WWW Services Master properties.

- 3 Select the Home Directory tab.
- 4 Under Application Settings, select the Configuration... button.
- 5 In the Application Configuration window, select the App Mappings tab.
- 6 If the .lasso extension does not appear under Application Mappings, select the Add button.
- 7 Next to the Executable field, select the Browse button and navigate to the Windows\System32 directory, then select Lasso8ConnectorforIIS.dll.
- 8 Enter .lasso in the Extension field.
- 9 Under Verbs, select the All Verbs radio button.
- 10 Check the Scripts engine check box.
- 11 Select OK to apply the changes.
- 12 Repeat steps 6 through 11 for the .LassoApp extension.
- 13 Right-click on the computer name in the Internet Services Manager and select Restart IIS.

File Extensions Note: Lasso can be configured to process file extensions other than .lasso or .LassoApp by entering a different extension name in step 6. However, the .lasso and .LassoApp extension must be configured, as not doing so will prevent access to required files for Lasso Professional Server, such as the administration LassoApp.

Remote IIS Configuration

This section describes setting up Lasso Professional Server for a distributed architecture where Lasso Service runs on a separate Windows computer from the IIS Web server. The first step in setting up a distributed architecture is copying the necessary files to the remote Web server, and disabling the Lasso Web server connector on the local machine. To do this, follow the procedure below.

To setup Lasso Connector for IIS on a remote IIS Web server:

- 1 On the machine running Lasso Service, stop Lasso Service using the Services menu in Windows.
- 2 On the machine running Lasso Service, stop IIS using the Services menu in Windows. Since the Lasso Connector for IIS is being moved to a remote machine, all Lasso pages and LassoApps (including Site Administration) should no longer be served from this machine.
- 3 On the machine running Lasso Service, locate the Lasso8ConnectorforIIS.dll file, located in the Windows\System32 folder or WinNT\System32 folder (depending on the system), and move it to the Windows\System32 folder or WinNT\System32 folder on the remote IIS Web server.
- 4 On the machine running Lasso Service, locate the Lasso8ISAPI.conf file, located in the Windows\System32 folder or WinNT\System32 folder (depending on the system), and move it to the Windows\System32 folder or WinNT\System32 folder on the remote IIS Web server.
- 5 On the remote IIS server, configure IIS for Lasso Professional Server as specified in the previous *Lasso Connector for IIS* section.
- 6 On the remote server, open the Lasso8ISAPI.conf file located in the Windows\System32 folder or WinNT\System32 folder in a plain text editor.
- 7 Next to Server, replace 127.0.0.1 with the correct host name or IP address of the computer running Lasso Service.
- 8 On the machine running Lasso Service, start Lasso Service using the Services menu.
- 9 On the remote Web server, restart IIS using the Services menu.

Note: In Lasso Professional Server, only one Web server may connect to the machine running Lasso Service at a time.

Setting Up Redundant Lasso Professional Server Servers

For load balancing purposes, multiple Web servers running Lasso Professional Server may be set up in a RAIC (Redundant Array of Independent Components) in the same manner as other Web servers. There are a variety of ways to do this depending on the networking hardware and software available, and instructions for setting up a RAIC are beyond the scope of this guide.

However, there is one setting in Lasso Professional Server that must be adjusted in order for Lasso sessions to persist across a RAIC, and that is to point each instance of Lasso Professional Server to a single MySQL sessions server. For instructions on how to do this, please see the *Setting Site Preferences* chapter.

Installation Contents

This section contains a list of all files and folders that are installed during Lasso Professional Server installation.

System User Note: The Lasso Professional Server installer automatically configures Lasso Service under the System user account in Windows. The privileges of this user are invoked whenever Lasso needs to read, write, execute, or create files or folders on the hard drive.

Start Menu

Lasso Professional Server installs the Uninstaller application in the Start Menu\OmniPilot Software\Lasso Professional Server folder in the Windows hard drive. The Uninstaller application allows one to make changes to an existing installation of Lasso Professional Server, or remove Lasso Professional Server from the system.

Program Files

Lasso Professional Server installs the following files and folders in the Program Files\OmniPilot Software\Lasso Professional 8 folder in the Windows hard drive.

- **Chart FX folder** – Contains additional documentation about the Chart FX module and the Chart FX for Java Designer which can be used to create Chart FX template files.
- **Documentation folder** – Contains all Lasso Professional Server documentation in PDF format, which can be viewed with Adobe Acrobat Reader. The following files and folders are contained in this folder. It is highly recommended that the documentation be read in this order.
 - 1-ReadMeFirst folder – Contains Release Notes.txt and License Agreement.txt. Read this documentation in this folder first before proceeding to configure Lasso Professional Server.
 - 2-SetupGuide folder – Contains Lasso 8.5 Setup Guide.pdf.
 - 3-LanguageGuide folder – Contains Lasso 8.5 Language Guide.pdf, and the LassoApps, LCP, LCAPI, and LJAPI folders.
- **Extensions folder** – Contains additional optional components.
 - J2EE folder – Contains the Lasso Connector for J2EE which is required for Java Enterprise Server support.
 - Apache 2 folder – Contains the Lasso Connector for Apache 2 which is required for manual installation of support for Apache 2.
 - Apache 2.2 folder – Contains the Lasso Connector for Apache 2.2 which is required for manual installation of support for Apache 2.2.
 - OS Process folder – Contains the [OS_Process] module which allows access to the shell or other processes running natively on each platform.
- **ImageMagick Libraries** – The libraries for ImageMagick 6.2.9 are installed in the Lasso Professional 8 folder directly. These libraries are used to power the [[image] tags in Lasso 8. For more information, see the *Images and Multimedia* chapter in the Lasso 8.5 Language Guide.

- **Javalibraries folder** – Any extra Java libraries (usually .jar files) can be placed in this folder. They will be loaded when Lasso starts Java, but no JDBC drivers or tag modules will be loaded from them. JavaBeans can also be installed in this folder.
 - iText.jar – Required library for the [PDF_...] tags.
- **JDBCDrivers folder** – Drivers for JDBC data sources are placed here (empty by default). For instructions on how to use this folder while configuring JDBC data sources, see the *Setting Up Data Sources* chapter.
- **Lasso Admin folder** – Contains files and folders used for administering Lasso Professional Server.
 - BuildLassoApps folder – Is a folder used to build LassoApps. For more information, see the *Administration Utilities* chapter.
- **LassoApps folder** – Contains LassoApps for on-demand loading. When a LassoApp file is requested this folder is checked first and the LassoApp is loaded from here if it exists. This allows LassoApps to be served without placing them in the Web server root.
- **LassoLibraries folder** – Contains LassoApps and .Lasso pages for on-demand loading. Each file should have the name of the namespace it defines. When Lasso tries to execute an unknown tag it checks here to see if it can load the tag before continuing.
- **LassoModules folder** – Contains all default Lasso modules included with Lasso Professional Server as listed below. All data source connectors (including third-party connectors) and tag modules should go here.
 - chartfx.lasso.jar – Implements Chart FX support.
 - FMPConnector.dll – Data source connector module for FileMaker Pro data sources.
 - FMSAConnector.dll – Data source connector module for FileMaker Server data sources.
 - ImageTag.dll – Implements support for the image tags..
 - LDAPConnector.dll – Implements support for querying LDAP data sources.
 - LJAPI.jar – Implements core support for all LJAPI modules and includes the data source connector for JDBC data sources.
 - LJAPI.dll – Implements support for LJAPI modules.
 - MySQLConnector.dll – Data source connector module for MySQL data sources.
 - NSLookup.class – Java class file for [NSLookup] tag.
 - OpenBaseConnector.dll – Data source connector module for OpenBase data sources.
 - PDF.jar – Required library for the [PDF_...] tags.
 - SQLConnector.dll – Implements support for SQL-based data sources including Oracle, PostgreSQL, and Microsoft SQL Server.

Note: There is no separate module for the Lasso data source connector for SQLite. This is built into the core Lasso Service code.
- **LassoService.exe** – Lasso Service is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service is the Lasso Web Data Engine.
- **LassoStartup folder** – Contains the Startup.LassoApp file and sites folder, which are used by Lasso Service during startup. All Lasso pages and LassoApps which need to be launched during startup should be placed here. For information on how to use the LassoStartup folder, see the *Lasso pages* chapter of the Lasso 8.5 Language Guide. For additional discussion of the LassoStartup folder, see the Lasso 8.5 Language Guide.
- **SQLiteDBs folder** – Contains SQLite databases used to store Lasso Professional Server settings and preferences.

Web Server Root

Lasso Professional Server does not install any files in the Web server root. All of Lasso's administration tools are loaded on-demand from the LassoApps folder so do not need to be located in this folder.

Windows System Files

Lasso Professional Server installs the following files in the `Windows\System32` directory of the Windows hard drive. The files installed in this folder allow Lasso Professional Server to function properly. Lasso Web server connectors are also located here by default.

- **Lasso8ConnectorforIIS.dll** – The ISAPI filter that allows Lasso Service to communicate with Microsoft IIS version 5 and 6.
- **Lasso8ISAPI.conf** – The Web server configuration file for Lasso Connector for IIS. This file is not created by default.

Uninstalling Lasso Professional Server

Lasso Professional Server can be uninstalled using the uninstaller program provided in the Lasso Professional Server folder. Using the uninstaller program will automatically remove all core program and system files, and is the only recommended way to uninstall Lasso Professional Server.

Note: The uninstaller program will not remove any non-default files created after the time of install. See the following section for more information.

To uninstall Lasso Professional Server:

- 1 Select the Uninstall program, located in *Program Files > OmniPilot Software > Lasso Professional Server* in the Windows Start Menu. This launches the Lasso Professional Server Setup window.
- 2 Select the Remove radio button, and then select Next. Lasso Professional Server Setup displays a confirmation message for removing the application
- 3 Select Next in the confirmation screen. This will automatically uninstall Lasso Professional Server.
- 4 Select Finish when the uninstaller has completed.

Files and Folders Not Removed

The uninstaller program will not remove any non-default files created after the time of install, or any default Lasso files modified since the time of install. These files include the SQLite databases, Lasso log files, Lasso setups, and custom Lasso pages and LassoApps. These files and folders may be backed up for preservation, or may be deleted to completely remove Lasso Professional Server and all settings from the system.

- **Application folder** – The `C:\Program Files\OmniPilot Software\Lasso Professional Server` folder is left on the Windows hard drive, which contains the following folders:
 - JavaLibraries** – Contains any Java libraries installed by the administrator. This folder will be removed if no extra Java libraries were installed.
 - JDBCDrivers** – Contains any JDBC drivers installed by the administrator. This folder will be removed if no JDBC drivers were installed.
 - LassoApps** – Contains LassoApps for on-demand loading. This folder will be removed if no extra LassoApps were installed.
 - LassoLibraries** – Contains tag libraries for on-demand loading. This folder will be removed if no extra tag libraries were installed.
 - LassoModules** – Contains data sources connectors, Java libraries, and any third-party Lasso modules.
 - LassoSites** – Contains the site level folders for each defined site. The site level folders have the same structure as the master level folders described here. Each site might have its own JavaLibraries, JDBCDrivers, LassoApps, LassoLibraries, LassoModules, and SQLiteDBs.
 - SQLiteDBs** – Contains all default SQLite databases, which contain serial numbers, passwords, database settings, logs, and permissions from the previous setup of Lasso Professional Server.

- **Web Server Root folder** – The Web server root folder retains all Lasso pages created or modified since the time of installation. Customized Lasso pages, Web pages, and scripts will not be deleted during uninstallation.

Reinstallation Note: Any files remaining on the system from a previous installation of Lasso Professional Server will not be overwritten by installing a newer version. Therefore, all previous settings will be retained if a newer version of Lasso Professional Server is installed over the files not removed during the uninstallation of the previous version.

5

Chapter 5 Configuring on Linux

This chapter contains the installation and configuration instructions for Lasso Professional Server on Red Hat Enterprise Linux 4. It also contains information on Lasso Professional Server components and how they can be manually installed or modified.

- *System Requirements* lists the minimum system requirements for Lasso Professional Server.
- *Required Packages* lists the packages that Lasso Professional Server requires.
- *Web Browser Requirements* lists the Web browser requirements for administering Lasso Professional Server.
- *Installation Instructions* includes step-by-step instructions for installing Lasso Professional Server and establishing a basic setup.
- *Running Lasso Professional Server* describes how to start and stop Lasso Service, and how to check if it is running properly.
- *Extended Configuration* describes how to manually configure Apache for Lasso Professional Server, and how to set up a Lasso Professional Server distributed architecture.
- *Installation Contents* lists the files installed with Lasso Professional Server.
- *Uninstalling Lasso Professional Server* includes step-by-step instructions for removing Lasso Professional Server from your system.

System Requirements

Lasso Professional Server will run on systems which meet the minimum requirements listed below. Although Lasso Professional Server may run on machines which do not meet these requirements, these installations will not be supported.

- 300 MHz or higher Pentium-compatible CPU.
- 512 MB of RAM. More recommended. (Requires an additional 64 MB of Available RAM for each additional site. See the *Setting Up Sites* chapter for more information.)
- Standard installation of Red Hat Enterprise Linux 4 with network support.
- Red Hat Linux installation of Apache Web Server 2 (included with Red Hat Enterprise Linux 4).
- Sun's Java Runtime Environment (JRE) 6 for Red Hat Linux is required for Java-based tags and JDBC data sources to function. Installation instructions are included below.

Note: ImageMagick is automatically installed with Lasso Professional 8.5.2 and higher.

Certification Note: Lasso Professional Server is certified to work with Red Hat Enterprise Linux version 4 with the default installation of Apache Web server 2. Modified versions of any required components may work with Lasso Professional Server, but can not be officially supported.

Required Packages

Several packages are required in Red Hat Enterprise Linux 4 for all features in Lasso Professional Server to function. Many of these packages should be installed with server installations of Red Hat Enterprise Linux 4 by default. The required packages are summarized in the following table.

To view a list of packages that are required by the Lasso Professional Server installer, execute the following command in the terminal. The following command assumes you are currently in the same folder as the Lasso Professional Server installer.

```
rpm -qpR Lasso-Service-8.5.0-1.i386.rpm
```

If a package requirement is not met, then the `Lasso-Service-8.5.0-1.i386.rpm` file will indicate the packages that are missing upon installation. These packages may be installed from the Red Hat Enterprise Linux 4 installation CDs, or may be downloaded from URLs indicated in the release notes.

Required Packages

Lasso requires the following packages be installed before Lasso's RPMs are installed. All RPMs are provided by the Red Hat Enterprise Linux 4 installation CDs or the Red Hat Enterprise 4 Extras CD (Bonus Disc). The `java-1.4.2-ibm` RPMs in particular are provided on the Red Hat Enterprise Linux Extras CD.

Table 1: Lasso Service Required Packages

Package	Version
bzip2-libs	1.0.2
curl	7.12.1
e2fsprogs	1.35
freetype	2.1.9
glibc	2.3.4
krb5-libs	1.3.4
libgcc	3.4.3
libidn	0.5.6
libjpeg	6b
libstdc++	3.4.3
libtiff	3.6.1
libtool-libs	1.5.6
libxml2	2.6.16
libxslt	1.1.11
openssl	0.9.7a
pcre	4.5
xorg-x11-libs	6.8.1
zlib	1.2.1.2

Table 2: Lasso Connector for Apache 2 Required Packages

Package	Version	Notes
httpd	2.0.52	
libstdc++	3.4.3	
glibc	2.3.4	
libgcc	3.4.3	

See also the *Installation Instructions* that follow for notes about some specific packages that need to be installed.

Web Browser Requirements

This section describes the basic Web browser requirements needed to successfully set up and administer Lasso Professional Server in Red Hat Linux.

- Konquerer 3.0 or higher, FireFox, or compatible browser.
- Javascript enabled.
- Cookies enabled.
- Cascading Style Sheets support.
- Monitor capable of 800 x 600 resolution.

If these criteria are not met, then a browser check dialog box will be shown when trying to access Lasso Administration. You may proceed into the interface without meeting all these criteria, however some elements may not work properly. It is highly recommended that a browser that meets the listed requirements be used.

Installation Instructions

This section discusses installation procedures for Lasso Professional Server, and initializing Lasso Administration for the first run. This section will introduce the following:

- **Installing Java** describes installing the Java components necessary for Lasso Professional Server's Java-based features to function properly.
- **Installing Lasso Professional Server** describes how to install Lasso Professional Server using the auto-installation program.
- **Initialization** describes the initialization of Lasso Professional Server when running for the first time.

Installing Java

This section discusses installing Sun's Java Runtime Environment (JRE) 6 for Red Hat Linux. JRE 6 is required to successfully use the Java-based features available in Lasso Professional Server. These include LJAPI modules, JDBC data source connectivity, and Lasso tags which are dependent upon Java. For details on which Lasso tags are dependent upon Java, see the Lasso 8.5 Reference. It is not necessary to install Java if these features will not be used.

To install JRE 6:

- 1 Download Sun's JRE installer for Linux at the following URL. Download the "Linux RPM in self extracting file" e.g. `jre-6-linux-i586-rpm.bin`.

```
http://java.sun.com/javase/downloads/index.jsp
```

- 2 Use the following commands to decompress and install the RPM:

```
chmod a+x jre-6-linux-i586-rpm.bin
sudo ./jre-6-linux-i586-rpm.bin
sudo rpm -ivh jre-6-linux-i586-rpm.bin
```

The Java components needed to run the Java-based features in Lasso Professional Server are now installed. The Java-based features of Lasso Professional Server can be set up, enabled, and tested within Lasso Administration. See the *Setting Site Preferences* chapter *Java* section for more information.

Alternate JRE Versions

Lasso will look in the following location for the latest installation of Java. If a different version of Java is installed then a symlink can be created to select what version of Java Lasso finds. Lasso expects to find `libawt.so`, `libjava.so`, and `libverify.so` in this directory and `libjvm.so` in a `server` subdirectory.

```
/usr/java/latest/lib/i386/
```

Certification Note: While versions of Java other than JRE 6 may work with Lasso Professional Server, not all can be officially supported.

Installing Lasso Professional Server

The Lasso Professional Server installer for Red Hat Linux is distributed as three RPM files. These must be installed from the Red Hat Linux command line or console application.

- **Lasso-Service-8.5.0-1.i386.rpm** – This package will install the Lasso Service and Modules as all support files for Lasso Professional Server 8. This does not install the Web Server connector or the documentation for Lasso Professional Server.
- **Lasso-Documentation-8.5.0-1.i386.rpm** – This package will install the documentation and source files for LassoApps and Modules.
- **Lasso-Apache2Connector-8.5.0-1.i386.rpm** – This package will install the appropriate files and configure Apache 2 to connect to Lasso Professional Server 8.

The RPMs will install all required files for Lasso Professional Server, and will automatically start Lasso Service and Apache upon completion.

To install Lasso Professional Server:

- 1 Copy the three Lasso rpm files (`Lasso-Service-8.5.0-1.i386.rpm`, `LassoDocumentation-8.5.0-1.i386.rpm`, `Lasso-Apache2Connector-8.5.0-1.i386.rpm`) to the Red Hat Linux hard drive into the same folder. These files are unpacked from the `Lasso_Pro_8_Linux.tar.gz` archive along with the release notes.
- 2 Log in as the root user of the Red Hat Linux machine. This is done by entering `su` in the command prompt, then entering the root password. The hyphen loads the environment for the root user in the resulting shell.

```
su -
```

- 3 Install the Lasso packages. This can be accomplished using the following shortcut or with individual commands for each RPM file.
 - Enter the following command to install all three packages for Lasso Professional Server. The `rpm` command is called with the `-i` command to perform an install, the `-v` command for verbose feedback, and `-h` command to display a progress indicator.

```
rpm -ivh Lasso*.rpm
```

- Enter the following commands to install each package for Lasso Professional Server. The `rpm` command is called with the `-i` command to perform an install, the `-v` command for verbose feedback, and `-h` command to display a progress indicator.

```
rpm -ivh Lasso-Service-8.5.0-1.i386.rpm
rpm -ivh Lasso-Documentation-8.5.0-1.i386.rpm
rpm -ivh Lasso-Apache2Connector-8.5.0-1.i386.rpm
```

After Lasso Professional Server is installed and Lasso Service is successfully started, a message will appear with a URL address for initializing Lasso Professional Server in a Web browser. This is described in the next section.

Initialization

The Initialize Web page is where the administrator enters the Lasso Professional Server serial number, and sets the global administrator username and password for the first time.

To launch the Initialize page:

- 1 Launch a Web browser application. See the System Requirements section earlier in this chapter for Web browser requirements in Red Hat Linux.

Web Browser Note: Because Lasso Administration is Web-based, initialization and administration of Lasso Professional Server can be performed from a separate machine and operating system from the installation.

- 2 If on the same machine as the Lasso Professional Server installation, go to the following URL.

`http://127.0.0.1/ServerAdmin.LassoApp`

If on a different machine from the Lasso Professional Server installation, go to the following URL. Replace `www.example.com` with the domain name or IP address of the server running Lasso Professional Server.

`http://www.example.com/ServerAdmin.LassoApp`

Troubleshooting: In the event the Lasso Initialization page does not load, make sure that Lasso Service has started properly as described in the Running Lasso Professional Server section of this chapter. Otherwise, make sure that Apache is running, and that the installer was able to successfully edit the Apache configuration file (`httpd.conf`) as described in the *Extended Configuration > Lasso Connector for Apache* section of this chapter.

Initializing Lasso Professional Server

The Welcome to Lasso Professional Server panel displays a message listing the steps required to initialize Lasso. This involves entering a serial number, and setting the administrator username and password.

Figure 1: Server Initialize Page



For evaluation versions of Lasso Professional Server, an evaluation serial number is required to complete initialization. To request an evaluation serial number, select the Request Evaluation Serial Number button. This will take you to the LassoSoft Web site, where an evaluation serial number can be obtained. Once a serial number is obtained, Lasso Professional Server can be initialized by following the procedure below.

To initialize Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number in the Serial Number field. This must be entered exactly as it appears in your LassoSoft invoice using the following format:

`LP8.5-LIN-123456789`

Serial numbers are case-sensitive. Make sure that all capitalization is correct and that all letters and numbers are entered, including the LP8.5 prefix.

- 2 Enter your desired server administrator username in the Administrator Username field. This username will be used by the Lasso server administrator to log in to the administration interface, as discussed in the *Using Server Administration* chapter.

- 3 Enter your desired server administrator password in the Administrator Password field and re-enter the server administrator password in the Confirm Password field.
- 4 Select Submit Information to continue.

After the initialization information has been successfully submitted, an Initialization Complete panel is displayed stating that Lasso Professional Server has been successfully initialized.

Selecting the Open Server Administration button will open Server Administration, where server preferences can be established. Server Administration also provides access to the Site Administration interface for the default site.

Figure 2: Server Administration



At this point, Lasso Professional Server has been initialized and is ready to use. Lasso Professional Server provides a convenient, Web-based interface for configuring Lasso site settings, instantly managing and editing databases, configuring Lasso Security, monitoring events, and much more. It is within this interface that all Lasso settings and databases are set up and configured.

For a full description of Server Administration and how to use it, see the *Using Server Administration* chapter. For a description of Site Administration and how to use it see the *Using Site Administration* chapter.

The following describes where to go for more information on setting up and testing external data sources, and starting development.

- Setting up existing FileMaker Pro, MySQL, and JDBC data sources for use with Lasso Professional Server is described in the *Setting Up Data Sources* chapter.
- For an overview of setting up Lasso Security for databases, Lasso tags, files, and solutions, see the *Setting Up Security* chapter.
- To learn how to write custom Lasso solutions using Lasso code, see the Lasso 8.5 Language Guide.

Security Note: Lasso Professional Server uses ports 14552 for Lasso Service. Lasso Service cannot be directly accessed by anonymous users, however, it is recommended that system administrators block this port from incoming requests from IP addresses other than the IP address of the Web server using Lasso Professional Server.

Running Lasso Professional Server

Lasso Service is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Apache Web server. Lasso Service is started and stopped using the `lasso8ctl` command or the shell scripts provided in the `/usr/local/Lasso Professional 8/Tools` folder.

To start Lasso Service:

Log in as the root user of the Red Hat Linux machine. This is done by entering `su` in the command prompt, then entering the root password.

```
su -
```

- Use the `lasso8ctl` command to start Lasso Service.

```
/usr/sbin/lasso8ctl start
```

- Alternately, cd into the Tools folder and run the startLassoService.sh shell script.

```
cd "/usr/local/Lasso Professional 8/Tools"
./startLassoService.sh
```

Lasso Service can also be run in console mode within the terminal. This allows the administrator to view the Lasso Professional Server status window, which shows all Lasso Professional Server actions that are currently being performed, as well as actions that have been performed since start up. This is useful for monitoring and troubleshooting purposes. Lasso Service should be started this way whenever startup, stability, or performance problems are encountered in Lasso Professional Server.

To start Lasso Service in console mode:

Log in as the root user of the Red Hat Linux machine. This is done by entering su in the command prompt, then entering the root password. Then cd into the Tools folder and run the consoleLassoService.sh shell script.

```
su -
cd "/usr/local/Lasso Professional 8/Tools"
./consoleLassoService.sh
```

The terminal window contains all of the output of the Lasso Service console. Closing the shell will stop Lasso Service. To restart Lasso Service as a background service type Ctrl-C in the terminal to stop Lasso Service. Then type the following command.

```
/usr/sbin/lasso8ctl restart
```

To start Lasso Service:

Log in as the root user of the Red Hat Linux machine. This is done by entering su in the command prompt, then entering the root password.

```
su -
```

- Use the lasso8ctl command to stop Lasso Service.

```
/usr/sbin/lasso8ctl stop
```

- Alternately, cd into the Tools folder and run the stopLassoService.sh shell script.

```
cd "/usr/local/Lasso Professional 8/Tools"
./stopLassoService.sh
```

To see if Lasso Service is running:

Log in as the root user of the Red Hat Linux machine. This is done by entering su in the command prompt, then entering the root password.

```
su -
```

- Use the lasso8ctl command to check whether Lasso Service is running.

```
/usr/sbin/lasso8ctl status
```

- Alternately, run the following command to check for Lasso Service in the list of current processes.

```
ps -ax | grep Lasso8Service
```

Note: More than one Lasso8Service process will be shown due to the fact that Lasso Service is multi-threaded. Once Lasso Service is stopped using lasso8ctl or the stopLassoService.sh file, all Lasso8Service threads are killed.

Extended Configuration

This section discusses the changes the Lasso Professional Server installer makes to the Apache configuration file, and how to configure Lasso Professional Server for a distributed architecture with Apache. This section describes the following:

- **Lasso Connector for Apache** describes configuring the module that installs into Apache and provides connectivity to Lasso Service.
- **Remote Apache Configuration** describes manually configuring Lasso Service to run on a separate machine from Apache Web server (distributed architecture).
- **Setting Up Redundant Lasso Professional Server Servers** provides tips for setting up multiple Lasso Professional Server servers in a RAIC.

Important: It is not required that the instructions in this section be followed as part of a default installation or configuration process. The installer will configure a basic installation of Lasso Professional Server properly without any additional modification. This section is intended as an extended configuration reference for experienced administrators.

Lasso Connector for Apache

Lasso Connector for Apache is the default Web server connector included with Lasso Professional Server for Red Hat Linux. This section describes what settings Lasso Professional Server adds to the Apache configuration file (`httpd.conf`), and how to view or manually change these settings changes.

The Lasso Professional Server installer will automatically perform all necessary changes to the `/etc/httpd/conf/httpd.conf` file via an included `lasso.conf` file. After the altering the `httpd.conf` file to include the `lasso.conf` file, the installer makes a backup of the original file named `httpd.conf.lassobackup`. The following procedure describes viewing the changes the Lasso Professional Server installer makes to the Apache configuration file.

To view changes made to the Apache configuration file:

- 1 Log in as the root user of the Red Hat Linux machine. This is done by entering `su` in the command prompt, then entering the root password.

```
su -
```

- 2 At the command prompt, enter the following to open the `lasso.conf` file in the VI text editor.

```
vi /etc/httpd/conf/lasso8.conf
```

- 3 This file should contain the following lines:

```
LoadModule lasso8_module modules/Lasso8ConnectorforApache2.so
AddHandler lasso8-handler lasso
AddHandler lasso8-handler lassoapp
```

The `AddHandler` lines tell Apache to process files with the extensions `.lasso` and `.lassoapp` with Lasso Service.

File Extensions Note: Apache can be set up to process alternate file extensions with Lasso Service by adding additional lines using the following format: `AddHandler lasso-handler extension`. Replace extension with the file extension you wish to be processed by Lasso Service (e.g. `htm`, `html`, etc.).

- 4 Save the `lasso8.conf` file.
- 5 Restart Apache Web Server for any changes to take place. This is done by entering the following at the command prompt:

```
/usr/sbin/apachectl restart
```

Remote Apache Configuration

This section describes setting up Lasso Professional Server in a distributed architecture where Lasso Service runs on a separate Red Hat Linux machine from the Red Hat Linux Apache Web server. To do this, follow the procedure below.

To set up Lasso Connector for Apache on a remote server:

- 1 On the machine running Lasso Service, stop Lasso Service using the stopLassoService.sh file.
- 2 On the machine running Lasso Service, stop Apache if it is running. Since the Lasso Connector for Apache is being moved to a remote machine, all Lasso pages and LassoApps (including Lasso Administration) should no longer be served from this machine.
- 3 On the machine running Lasso Service, move the Lasso folder from the Web server root folder (e.g. /usr/www/html) to the Web server root folder on the remote server.
- 4 On the machine running Lasso Service, locate the /etc/httpd/conf/lasso8.conf file and move it into the /etc/httpd/conf/ folder on the remote server.
- 5 On the machine running Lasso Service, locate the /etc/httpd/modules/Lasso8ConnectorforApache2.so file and move it into the /etc/httpd/modules folder on the remote server.
- 6 On the machine running Lasso Service, locate the /etc/lasso/lassoapache.conf file and move it into the /etc/lasso folder on the remote server.
- 7 On the remote Web server, log in as the root user of the Red Hat Linux machine. This is done by entering su in the command prompt, then entering the root password.

```
su -
```
- 8 At the command prompt, enter the following to open the httpd.conf file in the VI text editor:

```
vi /etc/httpd/conf/httpd.conf
```
- 9 Add the following line to the end of the Apache configuration file (httpd.conf):

```
Include /etc/httpd/conf/lasso.conf
```
- 10 Save and close the httpd.conf file.
- 11 At the command prompt, enter the following to open the lassoapache.conf file in the VI text editor:

```
vi /etc/lasso/lasso8apache.conf
```
- 12 Next to Server, replace 127.0.0.1 with the correct host name or IP address of the machine running Lasso Service.
- 13 Save and close the lasso8apache.conf file.
- 14 On the remote server, restart Apache Web Server. This is done by entering the following command in the terminal:

```
/usr/sbin/apachectl restart
```
- 15 On the machine running Lasso Service, start Lasso Service. This can be done using the lasso8ctl start command or the startLassoService.sh shell script described in the Running Lasso Professional Server section of this chapter.

Note: In Lasso Professional Server, only one Web server may connect to the machine running Lasso Service at a time.

Setting Up Redundant Lasso Professional Server Servers

For load balancing purposes, multiple Web servers running Lasso Professional Server may be set up in a RAIC (Redundant Array of Independent Components) in the same manner as other Web servers. There are a variety of ways to do this depending on the networking hardware and software available, and instructions for setting up a RAIC are beyond the scope of this guide.

However, there is one setting in Lasso Professional Server that must be adjusted in order for Lasso sessions to persist across a RAIC, and that is to point each instance of Lasso Professional Server to a single MySQL sessions server. For instructions on how to do this, please see the *Setting Site Preferences* chapter.

Installation Contents

This section contains a list of all files and folders that are installed during Lasso Professional Server installation.

Lasso User Note: The Lasso Professional Server installer creates a lasso (lowercase) user and group account in Red Hat Linux that is used to read, write, and execute files related to Lasso Professional Server. This user is installed automatically, and no configuration for this user is required. Do not attempt to configure or change the settings for this user as it will affect the functionality of Lasso Professional Server.

Application Files

Lasso Professional Server installs the following files and folders in the `/usr/local/` Lasso Professional 8 folder on the Red Hat Linux hard drive.

- **Chart FX folder** – Contains additional documentation about the Chart FX module and the Chart FX for Java Designer which can be used to create Chart FX template files.
- **Documentation folder** – Contains all Lasso Professional Server documentation in PDF format. The following files and folders are contained in this folder. It is highly recommended that the documentation be read in this order.
 - 1-ReadMeFirst folder – Contains `Release Notes.txt`, `License Agreement.txt`, and `Mac OS X Tips.pdf`. Read this documentation in this folder first before proceeding to configure Lasso Professional Server.
 - 2-SetupGuide folder – Contains `Lasso 8.5 Setup Guide.pdf`.
 - 3-LanguageGuide folder – Contains `Lasso 8.5 Language Guide.pdf`, and the `LassoApps`, `LCP`, `LCAPI`, and `LJAPI` folders.
- **Extensions folder** – Contains additional optional components.
 - J2EE folder – Contains the Lasso Connector for J2EE which is required for Java Enterprise Server support.
 - Apache 2 folder – Contains the Lasso Connector for Apache 2 which is required for manual installation of support for Apache 2.
 - Apache 2.2 folder – Contains the Lasso Connector for Apache 2.2 which is required for manual installation of support for Apache 2.2.
 - OS Process folder – Contains the `[OS_Process]` module which allows access to the shell or other processes running natively on each platform.
- **ImageMagick folder** – Contains an installation of ImageMagick 6.2.9, which is used to power the `[Image]` tags in Lasso 8. For more information, see the *Images and Multimedia* chapter in the Lasso 8.5 Language Guide.
- **JavaLibraries folder** – Any extra Java libraries (usually `.jar` files) can be placed in this folder. They will be loaded when Lasso starts Java, but no JDBC drivers or tag modules will be loaded from them.
 - `iText.jar` – Required library for the `[PDF_...]` tags.
- **JDBCDrivers folder** – Drivers for JDBC data sources are placed here (empty by default). For instructions on how to use this folder while configuring JDBC data sources, see the *Setting Up Data Sources* chapter.
- **LassoAdmin folder** – Contains files and folders used for administering Lasso Professional Server.
 - `BuildLassoApps` folder – Is a folder used to build LassoApps. For more information, see the *Site Administration Utilities* chapter.

Lasso8Service.sh – This is the “watchdog” script for Lasso Professional Server. This file continuously queries Lasso Service to make sure that it is running when it is supposed to be, and automatically restarts Lasso Service if it inadvertently quits. When it does this, a log entry is added to the standard system log file (traditionally `/var/log/system.log`) in Mac OS X. If the number of restarts exceeds 20 per hour, the script stops attempting to start Lasso Service until the problem is fixed manually and the machine is rebooted.

Note: While Lasso Service is running, the `Lasso8Service.sh` file will temporarily place a file named `LassoService8.pid` in the `/tmp/` folder. This file is a helper file for `Lasso8Service.sh`.

- **LassoApps folder** – Contains LassoApps for on-demand loading. When a LassoApp file is requested this folder is checked first and the LassoApp is loaded from here if it exists. This allows LassoApps to be served without placing them in the Web server root.
- **LassoLibraries folder** – Contains LassoApps and .Lasso pages for on-demand loading. Each file should have the name of the namespace it defines. When Lasso tries to execute an unknown tag it checks here to see if it can load the tag before continuing.
- **LassoModules folder** – Contains all default Lasso modules included with Lasso Professional Server as listed below. All data source connectors (including third-party connectors) and tag modules should go here.

`chartfx.lasso.jar` – Implements Chart FX support.

`FMPConnector.so` – Data source connector module for FileMaker Pro data sources.

`FMSAConnector.so` – Data source connector module for FileMaker Server data sources.

`ImageTag.so` – Implements support for the image tags..

`LJAPI.so` – Implements core support for all LJAPI modules and includes the data source connector for JDBC data sources.

`LDAPConnector.so` – Implements support for querying LDAP data sources.

`LJAPI.jar` – Implements core support for all LJAPI modules.

`MySQLConnector.so` – Data source connector module for MySQL data sources.

`NSLookup.class` – Java class file for [NSLookup] tag.

`OpenBaseConnector.so` – Data source connector module for OpenBase data sources.

`PDF.jar` – Required library for the [PDF_...] tags.

`SQLConnector.so` – Implements support for SQL-based data sources including Oracle, PostgreSQL, and Microsoft SQL Server.

Note: There is no separate module for the Lasso Connector for SQLite. This is built into the core Lasso Service code.

- **Lasso8Service** – `Lasso8Service` is the core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service is the Lasso Web Data Engine.
- **LassoSites folder** – Contains the site level folders for each site that has been created. By default contains a default-1 folder for the default site. This folder contains most of the same folders as the Lasso Professional Server application folder (called the master level). In general, the site level is checked before the master level when Lasso needs to find a resource.
- **LassoStartup folder** – Contains the `Startup.LassoApp` file and `sites` folder, which are used by Lasso Service during startup. All Lasso pages and LassoApps which need to be launched during startup should be placed here. For information on how to use the LassoStartup folder, see the *Lasso pages* chapter of the Lasso 8.5 Language Guide. For additional discussion of the LassoStartup folder, see the Lasso 8.5 Language Guide.
- **SQLiteDBs folder** – Contains SQLite databases used to store Lasso Professional Server settings and preferences.
- **Tools folder** – Contains Mac OS X terminal command files for performing various tasks with Lasso Professional Server. These command files allow users to perform tasks using the terminal by simply double-clicking on the command files.

`consoleLassoService.sh` – Starts Lasso Service as an application in the terminal. Useful for viewing the startup log and troubleshooting. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

`startLassoService.sh` – Starts Lasso Service via the terminal. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

`stopLassoService.sh` – Stops Lasso Service via the terminal. For instructions on how to use this file, see the *Running Lasso Professional Server* section of this chapter.

Startup Files

The `/usr/sbin/` folder contains files that are used to auto-start Lasso Service at boot time.

- `lasso0ctl` – Auto-starts Lasso Service at boot time. A symbolic link to this file is also placed in `/etc/rc.d/init.d/`.

Web Server Root

Lasso Professional Server does not install any files in the Web server root. All of Lasso's administration tools are loaded on-demand from the `LassoApps` folder so do not need to be located in this folder.

System Files

Lasso Professional Server installs the following files in various system locations on the Red Hat Linux hard drive.

- **`httpd.conf`** – Lasso Professional Server modifies the Apache configuration file (`httpd.conf`) located in the `/etc/httpd/conf` folder.
- **`lasso8apache.conf`** – The Web server configuration file for Lasso Connector for Apache, located in the `/etc/lasso` folder.
- **`Lasso8ConnectorforApache2.so`** – This file is the Lasso Web server connector for Apache, located in `/etc/httpd/modules` folder.
- **`Lasso8Service.pid`** – This file is placed in the `/tmp` folder while Lasso Service is running.

Uninstalling Lasso Professional Server

Lasso Professional Server can be uninstalled using the procedure below. This will automatically remove all core program and system files, and is the only recommended way to uninstall Lasso Professional Server.

Warning: The RPM will remove all files installed by the RPM, including the `lasso_internal` database which stores all Lasso Administration settings. For preservation of settings, it is recommended that these database be backed up before uninstalling.

To uninstall Lasso Professional Server:

- 1 Log in as the root user of the Red Hat Linux machine. This is done by entering `su` in the command prompt, then entering the root password.

```
su -
```

- 2 Stop Lasso Service using the `lasso0ctl` command as follows.

```
/usr/sbin/lasso0ctl stop
```

- 3 Uninstall the Lasso packages. This can be accomplished using the following shortcut command to remove all Lasso packages or by removing the three Lasso packages individually.

- Enter the following command to uninstall all three packages for Lasso Professional Server. This command lists all installed rpm files, filters the list for those that contain Lasso, and then pipes the list to the rpm program with the remove option `-e`.

```
rpm -qa | grep Lasso | xargs rpm -e
```

- Enter the following three commands to remove each Lasso package individually. Note that the remove command does not require the version number for each package to be specified.

```
rpm -e Lasso-Service  
rpm -e Lasso-Documentation  
rpm -e Lasso-Apache2Connector
```

Files and Folders Not Removed

The RPM will not remove any non-default files created after the time of install. These files include new SQLite databases, Lasso log files, Lasso setups, and custom Lasso pages and LassoApps. These files remain in the following locations:

- **Applications Folder** – The `/usr/local/Lasso Professional 8` folder is left on the Red Hat Linux hard drive, which contains the following folders:
 - Admin – Contains exported Lasso setup information, MySQL backup tables, exported text data, and LassoApp build files from Lasso Administration.
 - JDBCDrivers – Contains any JDBC drivers installed to Lasso Professional Server by the administrator.
 - LassoModules – Contains data sources connectors, Java libraries, and any customized Lasso modules.
 - SQLiteDBs – Contains any new SQLite databases created since the original installation.
- **Web Server Root Folder** – The `/var/www/html/` folder retains all Lasso pages created or modified since the time of installation. Customized Lasso pages, Web pages, and scripts will not be deleted during uninstallation.
- **Linux Folders** – The `/etc/lasso` folder remains on the Red Hat Linux hard drive.

These files and folders may be backed up for preservation, or may be deleted to completely remove Lasso Professional Server and all settings from the system.

Reinstallation Note: Any non-default files remaining on the system from a previous installation of Lasso Professional Server will not be overwritten by installing a newer installation. Therefore, any exported Lasso Administration settings and SQLite databases from a previous installation may be imported and used in a newer installation of Lasso Professional Server without having to move files.

6

Chapter 6 Upgrading

This chapter includes instructions for upgrading from an earlier version of Lasso Professional. The first section contains instructions for upgrading from any version of Lasso Professional 8 or 8.5. The remainder of the chapter discusses upgrading from Lasso Professional 7 or earlier.

- *Upgrading Lasso Professional 8 or 8.5* includes instructions for upgrading an existing Lasso Professional 8 or 8.5 installation to the most recent version.
- *Overview* includes information about what has changed in Lasso Professional Server and basic upgrade instructions.
- *Upgrading on Mac OS X* includes instructions for upgrading from Lasso Professional 5, 6, or 7 on Mac OS X.
- *Upgrading on Windows* includes instructions for upgrading from Lasso Professional 5, 6, or 7 on Windows 2000, Windows 2003 Server, or Windows XP Professional.
- *Upgrading on Linux* includes instructions for upgrading Lasso on Red Hat Enterprise Linux 4.
- *Upgrading From Earlier Versions* includes some notes about upgrading from earlier versions of Lasso Web Data Engine.

Note: Please consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

Upgrading Lasso Professional 8 or 8.5

Installation of Lasso Professional 8 or 8.5 can be upgraded according to the instruction in the installation chapters of this setup guide.

- **Mac OS X** – The Lasso Professional 8.5 installer can be used to install Lasso over the top of a previous installation of Lasso Professional 8 or 8.5. No extra steps are required.
- **Windows** – The previous installation of Lasso Professional 8.5 should be uninstalled prior to installing the new version. The previous version can be uninstalled by using the **Remove** option which is shown when launching the Lasso Professional 8.5 installer or by using the Add/Remove Programs control panel. No database or settings files will be removed by this procedure.
- **Linux** – The Lasso Professional 8.5 installer can be used to install Lasso over the top of a previous installation of Lasso Professional 8 or 8.5. No extra steps are required.

Overview

Lasso 8.5 includes many new features which are detailed in the *About Lasso 8.5* chapter in this guide. The following list includes some important new and changed features to consider when upgrading from a previous version of Lasso.

- **Backward Compatibility** – Every attempt has been made to ensure 100% backward compatibility with developer-created solutions from Lasso Professional 7 or 8. No syntax modifications should be required to run Lasso Professional 7 or 8 code in Lasso Professional Server. See the *Upgrading* section in the Lasso 8.5 Language Guide for full details
- **SQLite** – The internal data source in Lasso Professional Server has been changed to SQLite. Lasso MySQL is no longer supported. This means that any solution specific databases that were hosted in Lasso MySQL need to be migrated to an external installation of MySQL. Also, any solutions that relied on manipulating the internal security databases directly need to be updated for SQLite compatibility.
- **Multi-Site Architecture** – The new multi-site architecture of Lasso Professional Server has required modifications to the built-in tools such as Lasso Administration (now Server and Site Administration). Details about how to access these tools in Lasso Professional Server are included later in this guide.

The migration to SQLite and to the multi-site architecture require some manual steps in order to upgrade an existing Lasso Professional 5, 6, or 7 installation to Lasso Professional Server with all settings intact.

Note: Please consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

The basic upgrade procedure (elaborated on in the platform-specific sections that follow) is as follows. Note that several of the steps can be performed well in advance of installing Lasso Professional Server in preparation of the upgrade.

To upgrade to Lasso Professional Server:

- 1 If solution databases are served from Lasso MySQL then an external installation of MySQL should be installed and the Lasso Settings Migrator used to move the databases from Lasso MySQL to the external installation.
- 2 The Lasso Settings Migrator should be used to export the settings from Lasso Professional 5, 6, or 7. The resulting text file will be imported into Lasso Professional Server.

Important: The preceding steps can be performed at any time *prior* to installing Lasso Professional Server. These steps cannot be performed once Lasso Professional Server has been installed.

- 3 It is always recommended to make a full backup of the previous installation of Lasso. On Windows, the previous installation of Lasso must be uninstalled before Lasso Professional Server can be installed.
- 4 Install Lasso Professional Server and initialize with the serial number, server administration username and password, and SMTP settings. When prompted, select the exported settings file to set server settings and default site settings.
- 5 Install any additional modules, LassoApps, startup items, or JDBC drivers that were installed in the previous installation of Lasso. It may be necessary to restart the default site in order for Lasso to load these items.
- 6 Test each solution to ensure that they are accessing their data sources successfully and that all of their code is working properly. Although we strive for full backward compatibility it is essential that all upgraded solutions be tested thoroughly to ensure proper operation.

These procedures are covered in more detail in the platform-specific sections that follow. The final section in this chapter includes notes about upgrading from earlier versions of Lasso Web Data Engine.

Note: Please consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

Upgrading on Mac OS X

This section includes detailed instructions for upgrading from Lasso Professional 5, 6, or 7 to Lasso Professional Server on Mac OS X. Some of the steps in this procedure must be performed prior to installing Lasso Professional Server. For details about upgrading from earlier versions of Lasso Web Data Engine consult the final section in this chapter.

Preparation

These steps can be performed at any time prior to installing Lasso Professional Server. However, these steps cannot be easily performed once Lasso Professional Server has been installed.

If solution databases are served from Lasso MySQL then an external installation of MySQL should be installed and the Lasso Settings Migrator used to move the databases from Lasso MySQL to the external installation. The settings from Lasso Professional 5, 6, or 7 should be exported to a text file for import into Lasso Professional Server.

To migrate databases to an external installation of MySQL:

- 1 Download the latest version of MySQL from MySQL AB. There is a binary installation option for Mac OS X. Consult the licensing information to see if a commercial license of MySQL is required.

<http://www.mysql.com/>

- 2 Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to migrate any solution-specific databases from Lasso MySQL to the external installation of MySQL.

Note: Using the Lasso Settings Migrator should not necessitate any changes to the solution hosted in Lasso Professional 5, 6, or 7. The migration to an external installation of MySQL can happen well in advance of installing Lasso Professional Server and can yield speed and feature benefits.

To export settings from Lasso Professional 5, 6, or 7:

Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to export all of the settings from Lasso Professional 5, 6, or 7 into a text-based format that Lasso Professional Server can understand.

This text file will be required in the Installation and Configuration section below once Lasso Professional Server has been installed. The export can be performed at any time prior to installing Lasso Professional Server.

Backup

It is always recommended to make a full backup of the previous installation of Lasso before performing any upgrades.

Installation and Configuration

The following procedure should be followed to install and configure Lasso Professional Server.

To install Lasso Professional Server:

Run the Lasso Professional Server installer. If necessary, customize the installation by choosing the appropriate Web server connector. Consult the *Configuring on Mac OS X* chapter for full details.

The installer should automatically launch the Web browser with the Server Administration initialization interface.

To configure Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number and select a server administrator username and password. Continue to the next screen.
- 2 Set up the software update notification and SMTP server preferences. The SMTP preferences are not automatically imported from the prior installation.
- 3 Choose a site administration username and password and select the export file from the Lasso Professional 5, 6, or 7 installation to automatically import its settings into the default site.
- 4 Continue to the final screen then on to Server Administration or Site Administration to continue configuring Lasso Professional Server if desired.
- 5 Install any additional modules, LassoApps, startup items, or JDBC drivers that were installed in the previous installation of Lasso. It may be necessary to restart the default site in order for Lasso to load these items.

Lasso Professional Server is now configured with all the settings from the Lasso Professional 5, 6, or 7 installation on the default site. No additional configuration should be necessary in order to run existing Lasso Professional 7 compatible Lasso pages on this server. Upgrades from Lasso Professional 5 or 6 may require some syntax changes. Consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

Testing

Test each solution to ensure that they are accessing their data sources successfully and that all of their code is working properly. Although we strive for full backward compatibility it is essential that all upgraded solutions be tested thoroughly to ensure proper operation.

Upgrading on Windows

This section includes detailed instructions for upgrading from Lasso Professional 5, 6, or 7 to Lasso Professional Server on Windows 2000, Windows Server 2003, or Windows XP Professional. Some of the steps in this procedure must be performed prior to installing Lasso Professional Server. For details about upgrading from earlier versions of Lasso Web Data Engine consult the final section in this chapter.

Preparation

These steps can be performed at any time prior to installing Lasso Professional Server. However, these steps cannot be easily performed once Lasso Professional Server has been installed.

If solution databases are served from Lasso MySQL then an external installation of MySQL should be installed and the Lasso Settings Migrator used to move the databases from Lasso MySQL to the external installation. The settings from Lasso Professional 5, 6, or 7 should be exported to a text file for import into Lasso Professional Server.

To migrate databases to an external installation of MySQL:

- 1 Download the latest version of MySQL from MySQL AB. There is a binary installation option for Windows. Consult the licensing information to see if a commercial license of MySQL is required.
<http://www.mysql.com/>
- 2 Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to migrate any solution-specific databases from Lasso MySQL to the external installation of MySQL.

Note: Using the Lasso Settings Migrator should not necessitate any changes to the solution hosted in Lasso Professional 5, 6, or 7. The migration to an external installation of MySQL can happen well in advance of installing Lasso Professional Server and can yield speed and feature benefits.

To export settings from Lasso Professional 5, 6, or 7:

Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to export all of the settings from Lasso Professional 5, 6, or 7 into a text-based format that Lasso Professional Server can understand.

This text file will be required in the Installation and Configuration section below once Lasso Professional Server has been installed. The export can be performed at any time prior to installing Lasso Professional Server.

Backup

It is always recommended to make a full backup of the previous installation of Lasso before performing any upgrades.

Uninstall Previous Version

It is necessary to uninstall the previous version of Lasso before installing Lasso Professional Server. This can be accomplished by launching the Lasso Professional 7 installer and choosing the remove option or using the Add or Remove Programs control panel. Uninstalling the previous version of Lasso Professional will not delete any of the settings databases or solution files.

Installation and Configuration

The following procedure should be followed to install and configure Lasso Professional Server.

To install Lasso Professional Server:

Run the Lasso Professional Server installer. If necessary, customize the installation by choosing the appropriate Web server connector. Consult the *Configuring on Windows* chapter for full details.

The installer should automatically launch the Web browser with the Server Administration initialization interface.

To configure Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number and select a server administrator username and password. Continue to the next screen.
- 2 Set up the software update notification and SMTP server preferences. The SMTP preferences are not automatically imported from the prior installation.
- 3 Choose a site administration username and password and select the export file from the Lasso Professional 5, 6, or 7 installation to automatically import its settings into the default site.
- 4 Continue to the final screen then on to Server Administration or Site Administration to continue configuring Lasso Professional Server if desired.
- 5 Install any additional modules, LassoApps, startup items, or JDBC drivers that were installed in the previous installation of Lasso. It may be necessary to restart the default site in order for Lasso to load these items.

Lasso Professional Server is now configured with all the settings from the Lasso Professional 5, 6, or 7 installation on the default site. No additional configuration should be necessary in order to run existing Lasso Professional 7 compatible Lasso pages on this server. Upgrades from Lasso Professional 5 or 6 may require some syntax changes. Consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

Testing

Test each solution to ensure that they are accessing their data sources successfully and that all of their code is working properly. Although we strive for full backward compatibility it is essential that all upgraded solutions be tested thoroughly to ensure proper operation.

Upgrading On Linux

This section includes detailed instructions for upgrading from Lasso Professional 5, 6 or 7 to Lasso Professional Server on Linux. Some of the steps in this procedure must be performed prior to installing Lasso Professional Server. For details about upgrading from earlier versions of Lasso Web Data Engine consult the final section of this chapter.

Preparation

These steps can be performed at any time prior to installing Lasso Professional Server.

If solution databases are served from Lasso MySQL then an external installation of MySQL should be installed and the Lasso Settings Migrator used to move the databases from Lasso MySQL to the external installation. The settings from Lasso Professional 5, 6, or 7 should be exported to a text file for import into Lasso Professional Server.

To migrate databases to an external installation of MySQL:

- 1 Download the latest version of MySQL from MySQL AB. There is a binary installation option for Red Hat Enterprise Linux. Consult the licensing information to see if a commercial license of MySQL is required.
<http://www.mysql.com/>
- 2 Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to migrate any solution-specific databases from Lasso MySQL to the external installation of MySQL.

Note: Using the Lasso Settings Migrator should not necessitate any changes to the solution hosted in Lasso Professional 5, 6, or 7. The migration to an external installation of MySQL can happen well in advance of installing Lasso Professional Server and can yield speed and feature benefits.

To export settings from Lasso Professional 5, 6, or 7:

Follow the instructions in the Lasso Settings Migrator (included in the Lasso Professional Server download package) to export all of the settings from Lasso Professional 5, 6, or 7 into a text-based format that Lasso Professional Server can understand.

This text file will be required in the Installation and Configuration section below once Lasso Professional Server has been installed. The export can be performed at any time prior to installing Lasso Professional Server.

Backup

It is always recommended to make a full backup of the previous installation of Lasso before performing any upgrades.

Installation and Configuration

The following procedure should be followed to install and configure Lasso Professional Server.

To install Lasso Professional Server:

Run the Lasso Professional Server installer. If necessary, customize the installation by choosing the appropriate Web server connector. Consult the *Configuring on Linux* chapter for full details.

The installer should automatically launch the Web browser with the Server Administration initialization interface.

To configure Lasso Professional Server:

- 1 Enter the Lasso Professional Server serial number and select a server administrator username and password. Continue to the next screen.
- 2 Set up the software update notification and SMTP server preferences. The SMTP preferences are not automatically imported from the prior installation.
- 3 Choose a site administration username and password and select the export file from the Lasso Professional 5, 6, or 7 installation to automatically import its settings into the default site.
- 4 Continue to the final screen then on to Server Administration or Site Administration to continue configuring Lasso Professional Server if desired.
- 5 Install any additional modules, LassoApps, startup items, or JDBC drivers that were installed in the previous installation of Lasso. It may be necessary to restart the default site in order for Lasso to load these items.

Lasso Professional Server is now configured with all the settings from the Lasso Professional 5, 6, or 7 installation on the default site. No additional configuration should be necessary in order to run existing Lasso Professional 7 compatible Lasso pages on this server. Upgrades from Lasso Professional 5 or 6 may require some syntax changes. Consult the *Upgrading* section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version.

Testing

Test each solution to ensure that they are accessing their data sources successfully and that all of their code is working properly. Although we strive for full backward compatibility it is essential that all upgraded solutions be tested thoroughly to ensure proper operation.

Upgrading From Earlier Versions

Lasso Professional Server does not automatically support upgrading from Lasso Web Data Engine or earlier versions of Lasso. This section includes some notes which should help make upgrading easier.

- Record the access information for each data source configured in Lasso including the IP address, username, and password. This will make it easy to reconfigure the data source in Lasso Professional Server.
- Lasso Professional Server does not support ODBC data sources or the local, AppleScript connection to FileMaker Pro on Macintosh. Installations which rely on these data sources will need to migrate to JDBC or the remote data source for FileMaker Pro.
- Lasso Professional Server does not have a CDML compatibility module. The Lasso CDML Converter available from LassoSoft's Web site can be used to convert CDML-based solutions to Lasso Professional Server Lasso syntax.
- Upgrades from Lasso Web Data Engine will require some syntax changes. Consult the Upgrading section of the Lasso 8.5 Language Guide for important information about changes to Lasso in this version and upgrade procedures.

The Lasso Updater in Lasso Studio can be used to automatically update sites that use an earlier version of Lasso to Lasso Professional Server preferred syntax.



Section III

Server Administration

This section describes the Server Administration interface in Lasso Professional Server including establishing server preferences, setting up sites, and using server utilities.

- *Chapter 7: Using Server Administration* describes how to access the Server Administration interface, provides an overview of the different sections of the interface, and introduces the user interface conventions it employs.
- *Chapter 8: Setting Server Preferences* including administrator settings, syntax settings, tag settings, and import/export of settings.
- *Chapter 9: Setting Up Sites* describes the server model of Lasso Professional Server including how to create and administer multiple independent sites.
- *Chapter 10: Server Utilities* describes the tools for managing and maintaining the Lasso Professional Server server including the SQL browser and error reporting.

Note: See the following section on *Site Administration* for details about configuring site preferences, setting up data sources, setting up Lasso security, and building and browsing databases.

7

Chapter 7

Using Server Administration

Lasso Professional Server provides a convenient, Web-based interface for configuring Lasso server settings, creating sites, checking for product updates, and much more. This interface is referred to as Server Administration.

This chapter provides an overview of Server Administration and is divided into the following sections.

- *Accessing Server Administration* describes how to connect to Server Administration through a Web browser
- *Exploring Server Administration* describes the three main sections of Server Administration.
- *Interface Elements* describes the interface elements and conventions used in Server Administration.
- *Lasso Internal Databases* provides an overview of the internal SQLite databases that power Server Administration and Lasso Professional Server.

Note: See the following section on *Site Administration* for details about configuring site preferences, setting up data sources, setting up Lasso security, and building and browsing databases.

Accessing Server Administration

Server Administration consists of the `ServerAdmin.LassoApp` file located in the `LassoApps` folder of the Lasso Professional Server application folder. Server Administration is a LassoApp file, which is accessed via a Web browser similar to a standard HTML page. The LassoApp is loaded on-demand by Lasso so does not need to be placed in your Web server root.

Server Administration uses the XHTML and CSS standards for rendering pages. Server Administration can be accessed in any Web browser that meets the Web browser requirements listed in the configuration chapters in this guide.

To access Server Administration:

In a Web browser, visit one of the the following URLs. Use `localhost` if accessing Server Administration from a browser on the same machine as Lasso Service. Otherwise, replace `www.example.com` with the actual domain name or IP address.

```
http://localhost/ServerAdmin.LassoApp
http://www.example.com/ServerAdmin.LassoApp
```

Note: On Linux machines it is necessary to specify the name of the LassoApp using proper case `SiteAdmin.LassoApp`. On Windows and Mac OS X the case of the file name doesn't matter.

Note: Web server rewrite technologies such as Apache `mod_rewrite` may interfere with loading Site Administration. An exception should be made for URLs with the `.LassoApp` extension in any rewrite rules.

The Web browser should prompt for the server administrator username and password. The Web browser will not prompt if Server Administration has already been accessed using the browser in the current session or if the authentication information has been stored in a keychain, passport, or browser preferences.

If an error is displayed, make sure Lasso Service and the Web server are running as described in the configuration chapters of this guide.

Figure 1: Server Administration



Note: The Server Administration LassoApp also includes the initialization page for Lasso. Consult the configuration chapters in this guide for instructions of how to initialize Lasso Professional Server.

Server Administrator Account

Only a person with the server administrator username and password may access Server Administration. The server administrator account is one of the first things that is set up after installing Lasso Professional Server, and is critical to the operation and administration of Lasso. The server administrator username and password should be protected since anyone who has it can make modifications to Lasso's server settings.

If for any reason the server administrator username and password is forgotten or lost, they can be reset using the procedure below. The Lasso Professional Server serial number is required in order to do this.

To reset a lost server administrator username and password:

- 1 Stop Lasso Service. On Mac OS X, type `sudo lasso8ctl stop` at the command line. On Windows, use the Services control panel to stop Lasso Professional Server 8. More details can be found in the appropriate configuration chapter in this guide.
- 2 Locate the `ResetAdminPassword.LassoApp` file in the Setup Guide documentation folder, and copy it into the `LassoStartup` folder. The Lasso Professional Server folder can be found in the Applications folder on Mac OS X or in the Program Files\OmniPilot Software folder on Windows.
- 3 Retart Lasso Service. On Mac OS X, type `sudo lasso8ctl start` at the command line. On Windows, use the Services control panel to start Lasso Professional Server 8. The `ResetAdminPassword.LassoApp` is automatically moved out of the `LassoStartup` folder.
- 3 Retart Lasso Service again. On Mac OS X, type `sudo lasso8ctl start` at the command line. On Windows, use the Services control panel to start Lasso Professional Server 8. This will ensure that Lasso will properly prompt for a new serial number.
- 4 Access Server Administration in a Web browser. This should cause the initialization screen to appear. Use `localhost` if you are accessing Server Administration on the same machine as Lasso Service. Otherwise, replace `www.example.com` with your actual domain name or IP address.

`http://localhost/ServerAdmin.LassoApp`
`http://www.example.com/ServerAdmin.LassoApp`

If the initialization screen does not appear then make sure that you restarted Lasso Service twice. The second restart is required for a full reset of Lasso’s administrator username and password, and serial number.

- 5 Enter your Lasso Professional Server serial number, desired username, and desired password in the initialization screen.

The Lasso server administrator username and password have been successfully reset and Server Administration may now be accessed normally. All of the sites, server preferences, and site settings on the machine will be unchanged.

Exploring Server Administration

This section describes the three main sections of the Server Administration interface, which are navigable via tabs at the top of the interface.

Note: See the following section on *Site Administration* for details about configuring site preferences, setting up data sources, setting up Lasso security, and building and browsing databases.

Setup

The Setup section allows the administrator to enter serial numbers, configure Chart FX, set up preferences for Lasso Professional Server, check for updates, and import or export Lasso’s server settings. The functionality of the Setup section is described in the *Setting Server Preferences* chapter.

Sites

The Sites section allows the administrator to add and configure individual Lasso sites. The functionality of the Sites section is described in the *Setting Up Sites* chapter.

Figure 2: Sites Section



Utility

The Utility section provides the server administrator various tools for maintaining the Lasso Professional Server server. This includes tools for executing SQL statements, monitoring errors, and more. The functionality of the Utility section is described in the *Server Utilities* chapter.

Figure 3: Utility Section



Support

The Support section provides links to various LassoSoft and third-party support resources.

Interface Elements

This section describes the basics of using and customizing the Server Administration interface.

Tab Navigation – Navigation occurs by selecting the text portion of a tab. Doing so will display the next tier of tabs with a page below. Each page generally contains two panels. The left panel provides a summary and listing of options and specified values. The right panel provides details for any selected list item. If a tab is disabled (i.e. does not link to a page upon selection), then one must fulfill a requirement such as selecting an option or button on a previous page before that page can be accessed.

Information Input And Display – All input within the interface is accomplished in the same manner as standard Web pages: by selecting HTML buttons, pull-down menus, check and boxes, and by entering information into text fields. A button name that ends in an ellipses (e.g. Databases...) means that the button will take the user to another page in Server Administration or to a dialog box. Feedback after performing an action within Server Administration occurs via a dialog box or bold-style text within a panel. Error messages are always in a red font.

Navigating Found Sets – Navigation within extended lists occurs via Prev and Next buttons, which are located under the last result in the page. Listings are always displayed in groups of ten, where selecting the Prev and Next buttons will display the previous or next ten results.

Status Bar – The status bar at the bottom of the screen shows the name of the current LassoApp. During an evaluation period, it will also show the number of days remaining in the evaluation period and a Buy Now link, which takes one to the LassoSoft online store. A Logout link is also provided allowing the current user to log out of Server Administration.

Lasso Server Databases

Lasso Professional Server uses several internal SQLite databases to store data and settings. These databases are required for Lasso Professional Server to function properly. This section describes the function for each of these databases.

- **lasso_global** – This is the database where all data that is input by an administrator into Server Administration is stored. This includes sites, site criteria, and server preferences defined in Server Administration.
- **site** – This database stores utility tables for the master site.
- **lasso_internal_#** – The internal databases for each site can be accessed through Server Administration. Each internal database is numbered with the site id.

None of the settings within these databases should be modified directly. All modifications should occur through the Server Administration interface. Making changes to these internal SQLite databases can have a severe negative effect on the performance and functionality of Lasso Professional Server.

8

Chapter 8 Setting Server Preferences

This chapter describes how to set server preferences for Lasso Professional Server. The **Setup > Server** section in Server Administration allows the Lasso server administrator to set server-wide preferences. These settings cannot be overridden by any site-specific or lower level security settings or via LassoScript.

- *Summary* provides an overview of Lasso Professional.
- *Serial Numbers* covers how to update Lasso's serial number.
- *Administrator Settings* describes how to change the administrator password or adjust Server Administration preferences.
- *Tag Settings* describes how to enable and disable Lasso tags for Lasso Professional.
- *Software Updates* describes how to check for updates to Lasso Professional.
- *Import/Export Settings* describes how to import and export Lasso 8 settings files.
- *Chart FX* describes how to activate Chart FX for Lasso.

Summary

The summary page provides an overview of Lasso Professional including the version, platform, and uptime. The panel on the left provides quick links to the settings and LassoApps for each defined site.

Figure 1: Summary Page



Serial Numbers

The serial number section allows the initialized serial number to be updated and displays information about the current serial number and software version.

Figure 2: Serial Number



To update the Lasso Professional Server serial number:

- 1 Enter or edit the serial number in the Serial Number field.
- 2 Select the Update Serial Number button.

Evaluation Note: If Lasso Professional Server is first initialized using an evaluation serial number and then you later purchase Lasso Professional Server, simply enter the purchased serial number into this form and select Update Serial Number. The Days Remaining message on the lower message bar will disappear after a valid purchased serial number is entered.

Administrator Settings

The administrator username and password is established when Lasso Professional Server is installed and initialized. Only the server administrator has the privileges to change Lasso's configuration or security settings, as described in the previous chapter.

The Admin Settings page is where the server administrator username and password can be changed, and other preferences can be established. The server administrator username and password are set during initialization as described in the configuration chapters in this guide.

The following preferences can be set.

- **Error Reporting** – A default Lasso Professional error page is shown whenever a .lasso or .lassoapp page is processed that contains an unhandled error. The Error Reporting pull-down menu allows the level of detail shown by the default Lasso Professional error page to be configured. The following levels may be set:
 - None – Sets the error page to show no information about the error that occurred. Only a message stating “An error has occurred” will be shown.
 - Minimal (Default) – Sets the error page to show only the error message and code number of the error that occurred. This is the default setting.
 - Full – All available information about the error (including source code) will be shown. This setting is recommended for development and debugging.
- **Digest Authentication** – Digest authentication is a more secure method of sending usernames and passwords from a Web browser to a Web server. If this setting is set to Enabled then Lasso will attempt to use digest authentication when possible.
- **Auto Refresh** – Controls whether administration screens auto refresh. The setting can be established for administration queues or administration errors pages.
- **Listing Size** – Sets the maximum number of items to show per page on listings pages.

Figure 3: Admin Settings Page



Updating the Server Administrator Password

The Administrator Settings panel allows the administrator username and password to be updated. The password is not shown, but can be changed by entering the current password with a new password and confirmation.

Lost Password Note: If the server administrator password is lost, please see the instructions at the beginning of the *Using Server Administration* chapter for resetting the password.

To change the server administrator username and password:

- 1 Enter an administrator username in the Username field. The username is ServerAdmin by default.
- 2 Enter the current administrator password in the Current Password field.
- 3 Enter a new password in the New Password field.
- 4 Enter the same password exactly as it was entered in the New Password field in the Confirm New Password field.
- 5 Select the Update Settings button.

After this change the browser will prompt to supply the new administrator username and password to continue using Server Administration.

Tag Settings

Lasso Professional Server provides server administrators the ability to restrict individual Lasso tags or tag categories from functioning within any site on the server. If a tag has been disabled, then an error will be displayed when it is called from within a Lasso page.

There are seven tag categories that may be restricted server-wide. For more information on these tags, see the Lasso 8.5 Language Guide.

- **Administration** – This category includes tags which return information about the environment of Lasso and allow security settings to be examined and changed.
- **Database** – Tags in this category allow the schema of MySQL databases to be altered.
- **File** – Tags in this category allow files on the local hard drive to be manipulated. Disabling this category will prevent Lasso from modifying any files on the local hard drive. It is recommended that you disable the [File_...] tags unless you are actively using them.
- **Include** – Tags in this category allow Lasso pages to be included in the current processing Lasso pages and for data from remote Web servers to be included and processed.
- **Technical** – The [Net] tags in this category can be disabled in order to disallow low-level connections with remote Web servers and other applications which are available over the Internet.
- **Utility** – The [Email_Send] and [Event_Schedule] tags can be disabled in order to prevent developers from sending email from Lasso or scheduling events for future execution.

Figure 4: Tags List Page



Tag Categories

The Tag Categories Listing panel shows a listing of the seven tag categories in LassoScript that contain tags that can be enabled or disabled. Tags defined in individual Lasso pages will not be listed here. A Mixed status indicates that a selected tag category contains both Enabled and Disabled tags. Selecting Enable All or Disable All will enable or disable all tags in each category shown.

Custom Tags Note: Third party tags cannot be enabled or disabled within Server Administration. Third party tags can only be disabled by removing the corresponding third party tag modules from the `Lasso Modules` folder, or by removing corresponding custom tag libraries from the `Startup.LassoApp` source and recompiling. A restart of Lasso Service is required for changes to take effect.

Updating Tags

When a category is selected in the Tag Categories Listing panel, it shows all tags in that category in the Tags Listing panel, which appears to the right. Each tag can be individually enabled or disabled by selecting Enabled or Disabled from the Status pull-down menu, and then selecting Update. All tags shown in the Tags Listing panel can also be enabled or disabled by selecting Enable All or Disable All.

Tag Dependencies

Some Lasso tags have dependencies on other tags to function properly. If a tag has been disabled that has dependencies with other tags, then those tags will not function correctly. To see whether or not a tag has any dependencies, consult the Lasso Reference and check to see if there are any tags listed in the Required field in the Detail view for any tag.

Software Updates

The Updates section in Server Administration automatically checks for updates to Lasso Professional and its components. If no updates are available then “Lasso is up to date” will be displayed. Otherwise, the panel will contain a list of available updates including a brief description and download link.

Import and Export Settings

The Import/Export page in Server Administration provides the ability to export the complete state of Lasso’s server configuration and security settings. Importing and exporting setup files is useful for backing up a particular Lasso Professional Server configuration, or setting up a new installation with the same configuration as another copy of Lasso Professional Server.

Important: This import/export section only handles server configuration settings. It does not import or export site-specific settings.

A Lasso setup file is an XML file that contains all custom server settings entered in Server Administration. The information in a Lasso setup file corresponds to the options selected in Server Administration.

Care should be taken with the setup files that are exported, as these files contain all of the security settings of Lasso Professional Server. Lasso stores passwords for users encrypted using MD5, but passwords for data sources may be stored in plain text. Export files which have been modified by an untrusted third-party should never be imported into Server Administration without careful review of the settings within the file.

Figure 5: Import/Export Page



Exporting Lasso Setup Files

The Export Setup panel allows the current Lasso Professional Server configuration to be exported as a text setup file to be preserved for future use.

To export a Lasso setup:

- 1 In the Export Setup panel, select the Export Setup button. This will automatically download a file that contains the contents your Lasso Professional Server setup file as Export_YYYYMMDDHHMMSS.txt.

Importing Lasso Setups

The Import Setup panel allows Lasso setup files to be imported. Only setup files exported using Lasso Professional Server or the Lasso Settings Migrator may be imported.

To import a Lasso setup:

- 1 In the Import Setup panel, select the Browse button, and choose the Lasso setup file on your hard drive you wish to import.
- 2 Select the Import Setup button. A warning informs the administrator that importing a setup will overwrite all existing installation settings with the settings being imported.
- 3 Select the OK button. This will overwrite all current settings with those stored in the imported setup file.
- 4 After the import has been performed Lasso Service must be restarted for the changes to take effect.

Note: While all current settings will be overwritten in an import, the settings provided during initialization (server administrator username and password, serial number, and registration information) will not be overwritten.

Chart FX

Software FX and LassoSoft have partnered to provide Chart FX for Lasso. This powerful charting package allows a diverse collection of charts to be created directly within Lasso. The Chart FX tags require a separate serial number from Lasso Professional. The Chart FX page in Lasso Server Administration allows the Chart FX tags to be activated. Without a serial number the Chart FX tags will run in developer mode which allows sample charts to be created so the functionality can be tried out before buying a serial number. More information about purchasing Chart FX for Lasso can be found at the LassoSoft store.

<http://store.lassosoft.com/>

Figure 6: Chart FX Page

**To Activate Chart FX for Lasso:**

Chart FX for Lasso requires a separate serial number from Lasso Professional. Visit the Setup > Chart FX > Serial Number section of Lasso Server Administration and enter the Chart FX for Lasso serial number. Select Activate Chart FX to automatically obtain an activation code. Once the activation is complete the Chart FX tags will be activated in all of Lasso's sites on this machine.

If the activation fails then a URL at Software FX will be displayed. Visit the URL and copy the provided activation code into the Activation Code field. After selecting the Activate Chart FX button the Chart FX tags will be activated in all of Lasso's sites on this machine.

9

Chapter 9 Setting up Sites

This chapter describes the server model of Lasso Professional Server including how to create and administer multiple independent sites.

- *Overview* provides an overview of the server model in Lasso Professional Server.
- *Creating Sites* describes how to create new sites including initialization of the new site.
- *Setting Up Sites* describes how to set site preferences and site selection criteria.
- *Site Preferences* describes how to modify the preferences for a site including the site administrator username and password and SMTP settings.
- *Allowed File Paths* describes how to establish what file paths users of a site can access.
- *Managing Sites* describes how to get information about site processes and troubleshoot problems.

Overview

Lasso Professional Server has a multi-site architecture. One master site (or process) accepts all incoming Web requests and routes them to the appropriate child site. This architecture allows different Web sites (e.g. for different clients or projects) to run in effect within their own copy of Lasso Service. Each site has its own settings, datasources, modules, sessions, and global variables. If one site crashes it can be restarted automatically without affecting any of the other sites.

- The **Master Site** is launched with Lasso Service and is responsible for the server-wide settings for Lasso. The master site routes all incoming Web requests to the appropriate child site. The master site is also responsible for the initialization of Lasso and for hosting `ServerAdmin.LassoApp`. The master site will automatically restart any child sites if they crash.
- The **Default Site** is created when Lasso is installed. The default site is set to receive all incoming Web requests. Lasso can be run with just the default site and master site without any further configuration. As additional sites are added the default site serves to catch any Web requests that are not served by any other site.
- Additional **Sites** (or child sites) can be created in `ServerAdmin.LassoApp`. Each site launches an additional Lasso Service process and can be configured completely separately from all other sites. If one child site crashes it can be restarted without affecting any of the other child sites.
- **Site Criteria** determines how an incoming Web request will be routed. Site criteria includes which Web host names (or IP addresses) should be routed to the site. Site criteria can also specify specific sub-folders within the Web root which should be routed to the site. Each site can have multiple criteria and criteria can have wildcards. This allows one site in Lasso to serve as the back-end for multiple Web hosts or even for multiple sites to serve as the back-end for different parts of the same Web host.

- Server Administration allows certain **Site Preferences** to be modified by the server administrator. These include the site administrator username and password, SMTP settings and more. These preferences can be set when a site is created or can be modified in the *Sites > Preferences* section.
- The server administrator also controls what **Allowed File Paths** a site has access to. The [File] tags will only work within the file paths specified in the *Sites > File Paths* section of Server Administration. The site administrator has full access within the specified file paths and can assign permissions for specific paths to groups created within the site. By default sites have access to the Web server root and the file uploads location.

System Requirements

Each site which is configured within ServerAdmin.LassoApp requires additional resources. By default Lasso launches the master site and default site. Combined, these two processes have comparable system requirements to earlier versions of Lasso.

Each additional site should have at least 64 MB of available RAM beyond that required for the basic Lasso installation.

Site Routing Criteria

The master site determines which child site to route incoming Web requests to using the site routing criteria. An incoming request has the following form:

`http://www.example.com/folder/file.lasso`

In this request `www.example.com` is the host name (or may sometimes be an IP address), `/folder/` is the path, and `file.lasso` is the particular file being requested. Lasso looks at the host name and path when determining where to route an incoming request.

How Lasso routes incoming requests based on site criteria:

- 1 The incoming host name is checked against each of the site routing criteria. The % symbol can be used as a wildcard within the criteria in order to match multiple host names. The default site has a host name of simply % which matches any incoming host name.
A site which should match for a specific site name might have a criteria of %example.com in order to match against `www.example.com`, `example.com`, `mail.example.com`, etc. A site can have multiple criteria if it needs to match multiple host names which can't be defined using a wildcard.
- 2 The incoming path is matched against the host root in each of the site criteria. The % symbol can be used as a wildcard within the criteria in order to match multiple paths. The default site has a host path of simply /% which matches any incoming path. Any sites which should match only against the host name should use the /% criteria for the host path.
A site which should match for a specific folder might have a criteria of /folder/% in order to match against `/folder/default.lasso`, `/folder/subfolder/default.lasso`, etc. A site might also have a criteria such as %/mail/% to match any paths that contain the folder mail. Or, a criteria of %MyLassoApp.LassoApp to serve a specific LassoApp from the site.
- 3 If multiple site criteria match an incoming request then the criteria with the longest host name and host path will be used. This means that the request will be routed to the site with most specific host name or host path criteria.

Note: The master site has built-in routing criteria. The Lasso initialization screen and ServerAdmin.LassoApp are automatically served from the master site.

Site Modules and Folders

The Lasso Professional Server application folder is divided into two levels. The master level includes all of the folders within the Lasso Professional Server application folder itself. The site level includes equivalent site-specific folders within the LassoSites folder.

The master level includes the following folders:

JavaLibraries	LassoModules
JDBCDrivers	LassoSites
LassoAdmin	LassoStartup
LassoApps	SQLiteDBs
LassoLibraries	

The LassoSites folder includes a subfolder for each site. By default there is one subfolder named `default-1` which corresponds to the default site. With the exception of the LassoSites folder, each site specific folder contains the same folders as the master level:

JavaLibraries	LassoLibraries
JDBCDrivers	LassoModules
LassoAdmin	LassoStartup
LassoApps	SQLiteDB

By default, all of the site level folders except SQLiteDBs are empty. SQLiteDBs contains the `lasso_admin`, `lasso_internal`, and `lasso_site_#` databases that hold the settings and preferences for the site.

Each site level folder contains a `LassoErrors.txt` file that records errors for the particular site based on the error routing preferences. The `LassoErrors.txt` file in the master level only records errors for the master site.

When Lasso requires a resource it looks in both the site level and the master level, but the order depends on the particular resource:

- **JavaLibraries** – All libraries which are included in the master and site levels are loaded when Java is started for each site. There should be no duplicate libraries between the site and master levels.
- **JDBCDrivers** – All drivers which are included in the master and site levels are loaded when Java is started for each site. There should be no duplicate drivers between the site and master levels.

Note: The master site does not load Java so no JavaLibraries or JDBCDrivers are loaded in the master site.

- **LassoAdmin** – The site level master admin is used for storage of backups, imports, exports, and created LassoApps. If a custom `error.lasso` page is located at the site level then it is used to format error messages, otherwise the default `error.lasso` file from the server level is used.
- **LassoApps** – This folder contains LassoApps that will be served on-demand. When a LassoApp is requested the site level is checked first. If the LassoApp is not found then the server level is checked.
- **LassoLibraries** – This folder contains on-demand tag libraries. When an unknown tag is called the site level folder is searched first for a file corresponding to the namespace of the unknown tag. If no tag library is found then the server level is checked.
- **LassoModules** – All modules in both the server level and site level folders are loaded when each site starts up. Modules in the server level folder are available to every site while modules in the site level folder are available only to one specific site.

Note: The master site does not load any modules.

- **LassoStartup** – All items in both the server level and site level folders are loaded when each sites starts up. The items in the server level folder are loaded first, followed by the items in the site level folder.

`Startup.LassoApp` is a special case. `Startup.LassoApp` is always loaded before the other startup items. By default there is one `Startup.LassoApp` in the master level folder. If `Startup.LassoApp` is located in the site level folder then it is loaded prior to loading any items in either the master or site level folders and the master level `Startup.LassoApp` is not loaded. This allows a site-specific `Startup.LassoApp` to override the default `Startup.LassoApp`.

Note: The master site only loads `Startup.LassoApp` from the master level and no other startup items.

- **SQLiteDBs** – Each site only has access to its own site level SQLite databases. The master site has access to the master level SQLite databases and also to all of the site level databases through special aliases. See the *Site Databases* section in *Managing Sites* for more details.

Site Independence

Each site within Lasso is completely independent from the other sites:

- **Settings** – Each site has its own preferences and settings which are established within SiteAdmin.LassoApp. The site administration can be accessed by either the site administrator or the server administrator. The server administrator can reset each site administrator's password.
- **Crashes** – If one site crashes, no other sites are affected. The master site will restart the site for the next incoming Web request that routes to the site.
- **Errors** – Each site maintains its own errors database in `lasso_internal` and its own `LassoErrors.txt` file within the site level folder.
- **Data Sources** – All data source hosts are set up for each site independently. This allows different sites to access completely different data sources or use different authentication information. The data sources which are available to the site include all modules and JDBC drivers within the site level and the master level folders.
- **Tags** – Each site has its own collection of tags. The available tags depend on what startup items and modules are loaded by the site. In addition, the server administrator can deactivate specific tags for every site on the server.
- **Groups and Users** – Each site has its own set of groups and users. All security settings for a group apply only within the site. Usernames only exist with the site and there is no automatic way to synch them between multiple sites.
- **Sessions** – Sessions can only be reloaded from within the same site they were stored. There is no way to access one site's sessions from another site.
- **Email Queue** – Each site is responsible for managing its own email queue. Pending messages can be seen from within SiteAdmin.LassoApp.
- **Scheduled Tasks** – Each site has its own scheduled events queue. However, since scheduled events are simply URLs it is possible for one site to call another (or a completely different Web site).
- **Global Variables** – Each site has its own set of global variables. There is no way to share global variables between site and it is impossible for users of one site to get access to global variables on another site.
- **Tag Libraries** – Each site can have its own collection of tag libraries which are loaded on-demand. Or, tag libraries which aren't found at the site level can be loaded from the master level.
- **LassoApps** – Each site has its own collection of loaded LassoApps. A LassoApp which is loaded on one site will not be available to any other sites. On-demand LassoApps can be loaded from the site level or from the master level.

Setting Up Sites

Sites are created in the *Sites > Sites* section of ServerAdmin.LassoApp. The default site is initially configured to catch all incoming Web requests. Many Lasso installations will only need to use this default site. Additional sites can be created for any of the following reasons:

- A Web hosting provider can create one site for each client on a server. This allows each client to use their own custom modules, startup items, data sources, security settings, global variables, etc. And prevents coding problems by one client to affect another client.
- A developer who hosts multiple sites may want to create one host for each site. This allows sessions and other settings to be applied to each site specifically.

- A site can be created for development purposes. This allows code which is being debugged to be tested separately from the deployment site. Once code is debugged and stable it can be moved over to the deployment site.

Figure 1: Sites List Page



The Site Listing panel shows a list of all the sites that are defined in Lasso. Each site includes its ID, name, enabled status, running status, and a link to the SiteAdmin.LassoApp for the site.

Selecting an individual site will show its settings in the right panel. Enable All will enable all defined sites. Disable All will disable all defined sites except for the default site. When a site is disabled it is also stopped.

Creating a Site

Selecting the Create...button will show the Add Site panel to the right.

Figure 2: Add Site Panel



To add a new site:

- 1 Enter a name of the site. Site names should consist of only alphanumeric characters and the underscore. The site level folders will be stored in a folder with the entered name, a dash, and the site ID. For example, New_Site_6.
- 2 Enter the site criteria. The host path matches against the host name or IP address of incoming Web requests. The % symbol can be used as a wildcard to match multiple host names. Host names are usually entered as %example.com to match www.example.com or example.com.

The host path should be `/%` to match any paths on the desired host. If the site should match only a specific subfolder of the desired host then the path can be entered as `/folder/%`.

- 3 The Description field can be used to store notes about the site.
- 4 The Duplicate Site popup can be used to optionally copy all of the site level folders from an existing site to serve as the basis of the new site. Otherwise, default settings will be used for the new site.
- 5 If Install Lasso Reference option is set to Yes then the LassoScript8_Reference will be copied into the site. This database is required in order to use the local copy of the Lasso Reference.
- 6 The Administrator Username and Password can be established for the new site. The site administrator will need this username and password to access the site's settings through SiteAdmin.LassoApp. The username cannot match that selected for the server administrator.

Note: The server administrator can also access each site's settings using SiteAdmin.LassoApp with the server administrator username and password.

- 7 The Prompt for Initialization option will prompt the site administrator to change their username and password and establish some other settings the first time they access SiteAdmin.LassoApp.
- 8 The SMTP settings allow the host, port, and SMTP AUTH username and password for the site to be established. By default, these settings match those which were set when Lasso was initialized.

Note: The site administrator username and password and site preferences can be modified on the **Sites > Preferences** section once the site is created.

After selecting the Add Site... button the newly added site is selected for additional modifications. If the site is enabled it will be started automatically. If the site is disabled it will not be started.

The following section describes the actions that Lasso takes when it creates a new site.

How Lasso creates a new site:

- 1 The site name and criteria are validated. A folder named with the site name and ID is created in LassoSites.
- 2 If a site has been selected for duplication then all of the site level folders for that site are copied to the new folder.
- 3 The Lasso Reference database is copied if required.
- 4 The site is started. Any additional required site level folders are created automatically. If the site level databases have not been duplicated then they are created automatically.
- 5 The site level lasso_internal database is accessed so the site administrator username and password and SMTP settings can be stored.
- 6 The default allowed file paths of the Web server root and file upload path are added to the site.

Updating Site Properties

Selecting a site name in the Sites Listing panel will show its details in the Site Detail panel which appears to the right. Here, the server administrator can update the sites properties and site criteria. Site preferences can be modified in the **Sites > Preferences** section and allowed file paths can be modified in the **Sites > File Paths** section.

To update details for a specific site:

- 1 The name and ID of the site are fixed after it is created. They cannot be modified.
- 2 A site can be enabled or disabled. If a site is disabled it will be stopped and will not be automatically started the next time Lasso is launched. Enabled sites are automatically started each time Lasso is launched.
- 3 The Description field can be used to store notes about the site.
- 4 The Site Admin URL is generated automatically based on the site criteria and is used for the Admin link in the site listing. The URL can be modified if required so it properly loads SiteAdmin.LassoApp for the site.

- 5 The Site Criteria are established in a listing each with a Host Name and Host Path. The % symbol can be used as a wildcard in either field. New site criteria can be added by inserting values into the New line then selecting Update. Site criteria can be deleted using the minus button next to each line.

All of the specified changes are made when the Update button is selected.

Stopping or Starting a Site

Selecting a site name in the Sites Listing panel will shows its details in the Site Detail panel which appears to the right. Here, the server administrator can start or stop the specified site.

To start a site:

Select the Start button at the bottom of the Site Detail panel. The Start button will only be visible if the site is currently stopped.

Note: Disabled sites can be started, but will not be automatically restarted if they crash. Disabled sites are not started automatically when Lasso is launched.

To stop a site:

Select the Stop button at the bottom of the Site Detail panel. The Stop button will only be visible if the site is currently running.

Note: Enabled sites can be stopped, but will automatically restart if an incoming Web request is routed to the site. Enabled sites will be started automatically when Lasso is launched.

To restart a site:

A site can be restarted by first stopping the site, then starting it again. An enabled site which is stopped will also automatically restart when an incoming Web request is routed to the site or if Lasso is relaunched.

Deleting a Site

Selecting a site name in the Sites Listing panel will shows its details in the Site Detail panel which appears to the right. Here, the server administrator can delete the specified site.

To delete a site:

Select the Delete... button within the Site Detail panel and confirm the dialog that appears. The site is stopped and deleted from within Lasso's settings.

The site level folders for the site are moved to a LassoSitesDeleted folder. To completely delete a site the corresponding site level folder must also be deleted from the LassoSitesDeleted folder.

Site Preferences

The *Sites > Preferences* section allows the site preferences to be modified. These include the site administrator username and password and the default SMTP settings for sending mail.

Figure 3: Site Preferences



The control at the top of the section allows the current site to be selected. Details of the current site are also provided in the left panel. Sites cannot be started and stopped through this interface. The *Sites > Sites* section should be used instead.

The *Prompt for Initialization* option controls whether the site administrator will be prompted for a new username and password the next time they log in to Site Administration. This allows the site administrator username and password to be set to a temporary value and then reset by the actual administrator to a value they choose.

Allowed Files Paths

Lasso has built-in file security for site users and site administrators. A site can only access the allowed file paths that are established in the *Sites > File Paths* section of Server Administration.

Figure 4: Allowed File Paths



The control at the top of the section allows the current site to be selected. Details of the current site are also provided in the left panel. Sites cannot be started and stopped through this interface. The *Sites > Sites* section should be used instead.

The left panel shows the file paths that are currently allowed for the site. The [File] tags will only work within a path that is defined in this panel for either the users or the site administrator of the current site. Paths can be deleted by selecting the minus symbol next to a path. Paths can be added by typing the name of the path in the text input and then selecting Add Path.

Important: The site administrator controls what permissions are assigned to a path. If a path is deleted and recreated within this interface then those permissions will be lost.

Several special purpose paths can be added automatically using the buttons at the bottom of the panel if they are not already included in the list of allowed paths. These include the Web server root path / and the file uploads path (established in the right panel).

File Uploads Path

The right panel allows the file uploads path to be established. This path can vary depending on platform, operating system version, local volume names, and Web server. The most foolproof way to determine the file uploads path is to try a file upload and then store the location of the uploaded temporary file.

The current file uploads path is displayed at the top of the panel. As long as this value is not Undefined it should not need to be redefined unless Lasso has been moved to a different machine. A new file uploads path can be determined by selecting a small (32k or less) file with the Choose File control and then selecting the Test button. The uploaded file will be deleted immediately after its location has been determined.

The path which is determined with this procedure can then be added to the list of allowed paths using the Allow File Upload Path button at the bottom of the left panel (if the path is not already in the list of allowed paths). Newly created sites will have this path allowed by default.

Managing Sites

This section includes technical details about sites in Lasso including how they are represented within system administration tools on each platform, what error messages they report, and how to troubleshoot problems.

Site Processes

Each individual site which is running in Lasso is seen as a separate process by the operating system.

- **Mac OS X** – Within the Activity Monitor or top or ps within the Terminal, each site will appear as a separate instance of Lasso Service. A default installation will consist of two instances of Lasso Service, one for the master site and one for the default site.

The following command can be used to list all of the site processes that are currently running. The command line arguments listed with each process help to identify which site each process corresponds to.

```
ps -U lasso
```

```
→ 11689 p1 S+ 0:03.50 ./Lasso8Service
   11690 p1 S+ 0:27.56 ./Lasso8Service--nolisten --ischild --siteid=1
      --sitename=default --pipename=lassosite_1 --workingdir=LassoSites/default-1
```

- **Windows** – Within the Services control panel only the master Lasso Service will be listed. The other site processes can be viewed by accessing the Windows Task Manager (Ctrl-Alt-Delete) and selecting the Processes tab. Each Lasso Service process will appear with image name LassoService.exe.

Error Messages

Lasso logs error messages related to site operations within the errors database and LassoErrors.txt file for the master site. Informative messages about the normal operation of sites are logged at the Warning level. Serious problems are logged at the Critical level.

- When the master site has to restart a site process the following message is logged as a Warning. This could indicate that the site process crashed or was restarted by an administrator.

```
Restarted site process <id> <name>.
```

- The rest of these errors are reported at the Critical level and only appear if a serious problem starting, connecting to, or restarting a site has been encountered. If one of these errors occurs the entire Lasso Service process should be stopped and then restarted.

Unable to hand request to site process even after restarting it <id> <name>.
Unable to restart site process <id> <name>.
Unable to get PID from child process for site <id> <name>.
Unable to create child process for site <id> <name>.
Child process failed to create named pipe for site <id> <name>.

10

Chapter 10 Server Utilities

This chapter documents the Utility section in Server Administration, which provides the administrator with tools for managing and maintaining the Lasso Professional Server server. This chapter describes the following:

- *SQL Browser* describes how to issue SQL queries to SQLite databases in the master site or specific child sites.
- *Errors* describes viewing and setting preferences for the Lasso error log.
- *Threads* describes how to view the threads that are currently running in Lasso.

SQL Browser

The SQL section provides a Web-based interface that allows one to issue SQL queries to the SQLite databases which store the settings for the master site or any of the child sites.

Issuing SQL Statements

The SQL Browser page is where SQL statements can be entered. The results of the query will be presented at the bottom of the page with one column for each field that is returned. The results could contain actual field data from a database search, or could contain synthetic field data which is generated by the database in order to format the results of various SQL statements and functions.

Figure 1: SQL Browser Page



To issue a SQL statement:

- 1 Select a site from the Site pull-down menu and a database from the Database pull-down menu.
- 2 Enter a SQL query in the SQL Statement field.

- 3 Select the Issue SQL Statement button. The results of the statement are shown in the lower panel, which by default shows No Statement Issues.

SQL Statement History

The last ten statements entered into the SQL Statement field will be displayed in a pull-down menu that appears to the right of History. In order to issue a statement stored in the History pull-down menu, simply select the statement from the pull-down menu, and it will automatically appear in the SQL Statement field. Then select Issue SQL Statement.

SQL Export

Selecting the Export area in the sidebar allows the results of a SELECT statement to be exported. Setting one or more options and then selecting Export Results will result in a text file being sent to the browser containing the current result set.

- **Format** – Specifies what format the export should be in. Options include Tab Delimited, Comma Delimited, SQL Inserts, or XML Data.
- **Field Names** – Specifies whether field names should be included as the first line of tab or comma delimited files. For XML data this preference controls whether field names are used as the tag names surrounding each field value or if a generic field container is used.
- **Line Endings** – Sets the line endings for the export to `\n`, `\r`, or `\r\n`.

SQL Options

Selecting the Options area in the sidebar allows the following options to be set:

- **Show Fields** – Allows the maximum number of fields displayed in the results to be selected. The options are 4 Fields, 8 Fields, 12 Fields, 16 Fields, and All Fields. If more than eight fields are selected, then the results panel will expand to the right to accommodate the extra fields.
- **Max Records** – Allows the maximum number of records displayed in the results to be selected. The options are 10 Records, 25 Records, 50 Records, 75 Records, 100 Records, and All Records.
- **Max Field Lines** – Allows the maximum number of lines of text to be displayed per field in the results to be selected. One line consists of approximately 20 characters horizontally. The options are 1 Lines, 4 Lines, 7 Lines, 10 Lines, and All Lines. Selecting All Lines will display the entire contents of the field.
- **Submit on Return** – If this option is set then a return in the SQL Statement field will submit the statement.

Errors

The Errors section allows the administrator to view and delete errors, warnings and details that have been logged both internally in Lasso Professional Server and via the [Log] tag, as well as set global logging options. For more information on the [Log] tag, see the *Files and Logging* chapter in the Lasso 8. Language Guide.

Important: Configuring error logging in Server Administration is not the same thing as configuring page-level error handling, such as for syntax errors and security errors. Page-level error handling is described in the *Error Reporting* chapter of the Lasso 8.5 Language Guide.

Lasso Errors

The Lasso Errors page lists all errors and system messages stored in the `lasso_global` database. Errors and system messages can also be deleted within this interface.

Figure 2: Errors Page



The Lasso Errors page contains a list of all error messages received by their date, time, level, and message. The list is navigable via the Prev and Next buttons in the sidebar.

Filtering Messages

The Level pull-down menu allows messages to be filtered by level, which includes Critical Errors, Warnings, Details, Action Statements, and Deprecated. Each error level is described individually in the following *Error Log Setup* section. Individual Lasso page and database errors are not logged here, but are rather controlled using the [Error...] tags. For more information, see the *Error Control* chapter in the Lasso 8.5 Language Guide.

Deleting Messages

Selecting one of the Delete All Errors button deletes all errors either in the current selected category or in the entire queue

Error Log Setup

The Setup page allows the administrator to set global logging options for Lasso Professional Server. These logging options are not limited to errors, and any feedback that occurs within the Lasso Service console can be logged for display in the Lasso Errors page.

Figure 3: Error Setup Page



The Error Setup panel allows the administrator to set unique log routing for the five log levels:

- **Critical Errors** – This includes fatal errors that affect the overall functionality of Lasso Professional Server. This can include startup errors, module and connector errors, and system-related errors that cause Lasso Professional Server not to function properly.

- **Warnings** – This includes non-fatal errors that do not affect the overall functionality of Lasso Professional Server, but the administrator should be aware of. This can include non-fatal startup errors, duplicate module or connector errors, SQL status errors, and logging errors.
- **Details** – This logs all processes performed by Lasso Service, which are not limited to errors.
- **Action Statements** – This logs all SQL statements generated by inline database actions.
- **Deprecated** – This logs warnings when deprecated tags or other functionality are used. Any deprecated tags or functionality may not be supported in a future version of Lasso.

For each feedback type, the administrator can choose to log or display feedback in any of the following three places:

- **Lasso Service Console** – If selected, feedback will be displayed in the Lasso Service console window when Lasso Service has been started as an application. For more information on starting Lasso Service as an application, see the configuration chapters of this guide.
- **Error Database** – If selected, feedback will be logged to the `_Errors` table in the `Lasso_Internal` database and displayed in the Lasso Errors page in Server Administration.

Note: The `_Errors` table will store a maximum of 10,000 records. Once this limit has been reached, the oldest records will automatically be deleted.

- **LassoErrors.txt** – If selected, feedback will be logged to the `LassoErrors.txt` file in the Lasso Professional Server folder in the hard drive.

Options are changed by checking or unchecking the desired values in the Error Setup panel and selecting the Update button. The Reset button returns all options to their default values. Selecting the Refresh button reloads the Setup page with its currently stored values.

Note: For information on how to programmatically log custom data to the logs described above, see the *Files and Logging* chapter in the Lasso 8.5 Language Guide.

Threads

The Threads page allows the threads that are currently running in Lasso Server Administration to be viewed. One thread always represents the page load which is generating the threads display. The only thread that will usually be running is the SQLite sessions manager.

Figure 4: Threads Page



IV

Section IV

Site Administration

This section describes the Site Administration interface in Lasso Professional Server including establishing site preferences, setting up datasources and hosts, setting up Lasso Security, and using site utilities.

- *Chapter 11: Using Site Administration* describes how to access the Site Administration interface, provides an overview of the different sections of the interface, and introduces the user interface conventions it employs.
- *Chapter 12: Setting Site Preferences* describes how to set preferences for a site in Lasso Professional Server including syntax, file extension, session, and Java settings.
- *Chapter 13: Setting Up Datasources* describes how to configure data sources for use with Lasso including configuring hosts, databases, tables, and fields.
- *Chapter 14: Setting Up Security* describes the methodology of securing Lasso-based Web sites using Lasso Security.
- *Chapter 15: Site Utilities* documents the utilities in Site Administration including the SQL browser, email queue, event queue, error logging, LassoApps, and cache control sections.
- *Chapter 16: Building and Browsing Databases* documents the Lasso Database Browser which allows browsing data from any Lasso-compatible database and building MySQL and SQLite databases.

Note: See the preceding section on *Server Administration* for details about configuring server preferences, setting up sites, and using server utilities.

11

Chapter 11

Using Site Administration

Lasso Professional Server provides a convenient, Web-based interface for configuring Lasso site settings, managing and maintaining databases, configuring Lasso Security, monitoring events, and much more. This interface is referred to as Site Administration.

This chapter provides an overview of Site Administration, and is divided into the following sections.

- *Accessing Site Administration* describes how to connect to Site Administration.
- *Exploring Site Administration* describes the three main sections of Site Administration.
- *Interface Elements* describes the interface elements and conventions used in Site Administration.
- *Lasso Internal Databases* provides an overview of the internal SQLite databases that power Site Administration and Lasso Professional Server.

Note: See the preceding section on *Server Administration* for details about creating sites and setting server preferences.

Accessing Site Administration

Site Administration consists of the SiteAdmin.LassoApp file which is loaded on-demand from the LassoApps folder in the Lasso Professional Server application folder. Site Administration is a LassoApp file, which is accessed via a Web browser similar to a standard HTML page.

Site Administration can be accessed in any Web browser that meets the Web browser requirements listed in the configuration chapters in this guide.

To access Site Administration:

Lasso Professional Server can have multiple independent sites defined within Server Administration. Each site handles incoming Web requests based on its site criteria which can be a mix of host name and path.

By default a single site is created that captures all incoming Web requests. If no additional sites have been created then Site Administration for the default site can be accessed using one of the following URLs. Use localhost if accessing Site Administration from a browser on the same machine as Lasso Service. Otherwise, replace www.example.com with the actual domain name or IP address.

```
http://www.example.com/SiteAdmin.LassoApp  
http://localhost/SiteAdmin.LassoApp
```

Note: On Linux machines it is necessary to specify the name of the LassoApp using proper case SiteAdmin.LassoApp. On Windows and Mac OS X the case of the file name doesn't matter.

Note: Web server rewrite technologies such as Apache `mod_rewrite` may interfere with loading Site Administration. An exception should be made for URLs with the `.LassoApp` extension in any rewrite rules.

To access Site Administration for a particular site it is necessary to construct a URL that meets the criteria for the site and add SiteAdmin.LassoApp. For example, if a site has site criteria specifying that the host name must be secure.example.com then the following URL would load Site Administration for that site.

```
http://secure.example.com/SiteAdmin.LassoApp
```

The status line at the bottom of Site Administration contains the name and ID of the site that is currently being accessed.

The Web browser should prompt for the site administrator username and password. The Web browser will not prompt if Site Administration has already been accessed using the browser in the current session or if the authentication information has been stored in a keychain, passport, or browser preferences.

If an error is displayed, make sure Lasso Service and the Web server are running as described in the configuration chapters of this guide.

Figure 1: Site Administration



Site Administrator Account

Only a person with the site administrator username and password (or the server administrator username and password) may access Site Administration. The site administrator account is set up by the server administrator and can be changed when the site is initialized.

If for any reason the site administrator username and password is forgotten or lost, they can be reset by the server administrator.

Exploring Site Administration

This section describes the three main sections of the Site Administration interface, which are navigable via tabs at the top of the interface. These sections are described below.

Note: The Build and Browse sections, formerly in Lasso Administration, have been moved to a new dedicated application called Lasso Database Browser, which is described in the *Building and Browsing Databases* chapter. Lasso Database Browser can be reached by using one of the quick links on the summary page.

Setup

The Setup section allows the administrator to set up data sources, enable and disable Lasso tags, set up Lasso security groups and users, import and export settings, and set site options for Lasso Professional Server.

Figure 2: Setup Section

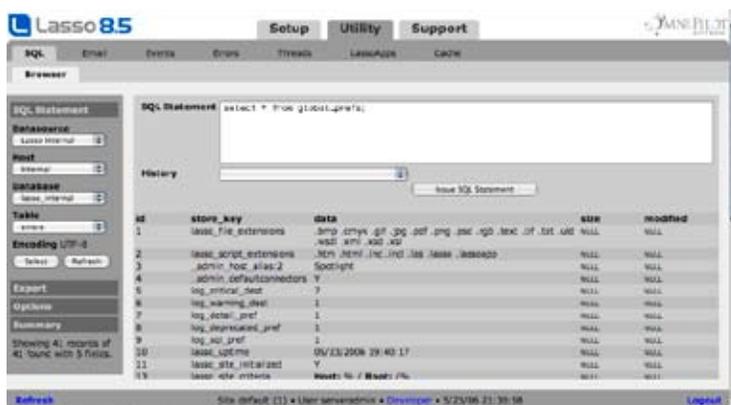


The functionality of the Setup section is described throughout this guide in the *Site Preferences*, *Setting Up Data Sources*, and *Setting Up Security* chapters.

Utility

The Utility section provides the administrator various tools for maintaining the Lasso Professional Server server. This includes tools for executing SQL statements, building LassoApps, monitoring URL event and email queues, monitoring errors, and more.

Figure 3: Utility Section



The functionality of the Utility section is described primarily in the *Administration Utilities* chapter.

Support

The Support section provides a launch pad for the Lasso Reference included with Lasso Professional Server, and provides links to various LassoSoft and third-party support resources.

The Lasso Reference is described in the *Lasso 8.5 Reference* chapter of the Lasso 8.5 Language Guide.

Figure 4: Support Section



Interface Elements

This section describes the basics of using and customizing the Site Administration interface.

Tab Navigation – Navigation occurs by selecting the text portion of a tab. Doing so will display the next tier of tabs with a page below. Each page generally contains two panels. The left panel provides a summary and listing of options and specified values. The right panel provides details for any selected list item. If a tab is disabled (i.e. does not link to a page upon selection), then one must fulfill a requirement such as selecting an option or button on a previous page before that page can be accessed.

Information Input And Display – All input within the interface is accomplished in the same manner as standard Web pages: by selecting HTML buttons, pull-down menus, check and boxes, and by entering information into text fields. A button name that ends in an ellipses (e.g. Databases...) means that the button will take the user to another page in Server Administration or to a dialog box. Feedback after performing an action within Server Administration occurs via a dialog box or bold-style text within a panel. Error messages are always in a red font.

Navigating Found Sets – Navigation within extended lists occurs via Prev and Next buttons, which are located under the last result in the page. Listings are always displayed in groups of ten, where selecting the Prev and Next buttons will display the previous or next ten results.

Status Bar – The status bar at the bottom of the screen shows the name of the current LassoApp. During an evaluation period, it will also show the number of days remaining in the evaluation period and a Buy Now link, which takes one to the LassoSoft online store. A Logout link is also provided allowing the current user to log out of Server Administration.

Lasso Site Databases

Lasso Professional Server uses several internal SQLite databases in which to store data and settings. These databases will appear in the Setup section of Site Administration with the status Enabled, and are required for Lasso Professional Server to function properly. This section describes the function for each of these databases.

- **lasso_internal** – This is the database where all data that is input by an administrator into Site Administration is stored. This includes data source information, group and user information, tag security information, serial numbers, and site preferences defined in Site Administration.
- **site** – Stores all site preferences, user data, queued email messages, scheduled events, and logged errors. These types of data are obtained via tags such [Session...], [Log...], [Email...], and [Event...], which are described in the Lasso 8.5 Language Guide.
- **lasso_admin** – This database stores preferences using by Site Administration and the Database Browser.

Due to the nature of the data in these databases, it is critical that these databases not be disabled. Making changes to these internal SQLite databases can have a severe negative effect on the performance and functionality of Lasso Professional Server.

12

Chapter 12 Setting Site Preferences

This chapter describes how to set site preferences for Lasso Professional Server. The *Setup > Site Settings* section in Site Administration allows the Lasso site administrator to set site preferences for the Lasso Professional Server server. These settings cannot be overridden by any lower level security settings or via Lasso.

- *Summary* covers password and serial number settings.
- *Administrator Settings* covers password and serial number settings.
- *Lasso Settings* describes how to change site settings for Lasso syntax, Lasso file extensions, and LassoApps.
- *File Extension Settings* describes how to change site settings for file extensions.
- *Session Settings* describes how to set site options for the [Session_...] tags in Lasso Professional Server.
- *Java Settings* describes how to configure Lasso to work with Java.
- *Import/Export Settings* describes how to import and export Lasso 8 settings files.

Summary

The Summary page provides an overview of the settings of the Lasso site. The left panel includes information about the version of Lasso, platform, uptime, etc. and a set of quick links to other site LassoApps. The right panel includes a summary of configured data sources, groups, and URL rewrite rules.

Figure 1: Summary Page

The screenshot displays the Lasso 8.5 Summary Page. The page is divided into several sections:

- Lasso Developer 8.5:**
 - Product: Lasso Developer 8.5
 - Version: 8.5.026
 - Platform: Mac OS X
 - Authentication: Basic
 - Uptime: 1 Hour
 - Site: default (1)
 - Site Criteria: Host: % / Root: /%
- Quick Links:**
 - Lasso Reference: [Reference LassoApp](#)
 - Database Browser: [DatabaseBrowser LassoApp](#)
 - Group Administration: [GROUPADMIN LassoApp](#)
 - Server Administration: [ServerAdmin LassoApp](#)
- Data Sources Summary:**

Name	Summary
Lasso Internal	3 Databases
FileMaker Pro	(Add Host)
FileMaker SA	(Add Host)
MySQL	(Add Host)
Spotlight	1 Database
SQL Server	(Add Host)
ODBC	(Add Host)
Oracle	(Add Host)
PostgreSQL	(Add Host)
- Security Summary:**

Name	Summary
Any User	All Users
OD Power Users	2 Users
Anonymous	1 Group
- URL Rewrite Summary:**

No Rules Found

Administrator Settings

The site administrator username and password is established when Lasso Professional Server is installed and initialized. Only the administrator has the privileges to change Lasso's configuration or security settings, as described in the previous chapter.

The Administrator Settings page is where the site administrator username and password can be changed. The site administrator username and password are initially set during initialization as described in the configuration chapters in this guide. The right panel also allows the following settings to be configured.

- **Auto Refresh** – Controls whether administration screens auto refresh. The setting can be established for administration queues or administration errors pages.
- **Listing Size** – Sets the maximum number of items to show per page on listings pages.

Figure 2: Admin Page



Updating the Site Administrator Password

The Administrator Settings panel allows the administrator username and password to be updated. The password is not shown, but can be changed by entering the current password with a new password and confirmation.

Lost Password Note: If the site administrator password is lost the server administrator can reset it. See the instructions in the *Server Administrator* section for more information.

To change the site administrator username and password:

- 1 Enter an administrator username in the Username field. The username is Administrator by default.
- 2 Enter the current administrator password in the Current Password field.
- 3 Enter a new password in the New Password field.
- 4 Enter the same password exactly as it was entered in the New Password field in the Confirm New Password field.
- 5 Select the Update Settings button.

During administrator login, the administrator will be prompted to supply the administrator username and password specified in the Administrator Settings panel.

To restart the site:

Select the Restart Site button. It can take up to a minute for the site to restart. No other sites in Lasso Service will be affected.

Lasso Settings

The Lasso Settings page in Site Administration allows the administrator to control how Lasso Professional Server processes certain types of Lasso syntax, and how it handles cases where certain types of code can adversely affect Lasso Service.

Figure 3: Lasso Settings Page



Syntax Settings

The Syntax Settings panel allows Classic Lasso syntax to be globally enabled or disabled, and also allows the Lasso Professional default error page to be configured.

- **Inline Host Specification** – This setting allows the use of inline host specifications to be disabled.
- **URL Rewrite** – This setting allows the URL rewrite system to be disabled.
- **Classic Lasso Syntax** – Classic Lasso syntax is a deprecated programming method which provides Lasso database commands within URLs or HTML forms. These commands are passed from the page in which they are specified to a response page which formats the results of the commands. Classic Lasso syntax is enabled by default to ensure compatibility with solutions built using earlier versions of Lasso.

Enabled – Allows Classic Lasso parameters to be used in URLs.

Minimal – Allows only -Response and -Error parameters to be used in URLs. This setting is equivalent to disabling Classic Lasso in earlier versions of Lasso.

Disabled – No Classic Lasso parameters including -Response and -Error are recognized. This is the most secure setting if you don't use Classic Lasso and is the recommended setting.

If Classic Lasso syntax is disabled, then older Lasso solutions using the Classic Lasso database methodology will not work. Disabling Classic Lasso enhances the security of a Lasso powered Web site by allowing database schema to be hidden and preventing site visitors from manufacturing URLs that perform different database actions than those explicitly programmed into the Web site.

- **Error Reporting** – A default Lasso Professional error page is shown whenever a .lasso or .lassoapp page is processed that contains an unhandled error. The Error Reporting pull-down menu allows the level of detail shown by the default Lasso Professional error page to be configured. The following levels may be set:

None – Sets the error page to show no information about the error that occurred. Only a message stating “An error has occurred” will be shown.

Minimal (Default) – Sets the error page to show only the error message and code number of the error that occurred. This is the default setting.

Full – All available information about the error (including source code) will be shown. This setting is recommended for development and debugging.

Note: Errors and error codes are further discussed in the *Error Control* chapter of the Lasso 8.5 Language Guide. A list of error messages and code numbers can be found in *Appendix B: Error Codes* of the Lasso 8.5 Language Guide.

- **LassoApps** – This setting allows the use of LassoApps to be disabled.
- **Digest Authentication** – Digest authentication is a more secure method of sending usernames and passwords from a Web browser to a Web server. If this setting is set to Enabled then Lasso will attempt to use digest authentication when possible.

Default Page Encoding

The Default Page Encoding panel allows the administrator to specify the default character set that Lasso Service should assume when interpreting data sent via client browsers (such as HTML form data). This can be UTF-8 (Unicode) or Latin-1 (ISO-8859-1). Unicode is used by default.

To set the default character set:

- 1 Select UTF-8 (Unicode) or Latin-1 (ISO-8859-1) from the Encoding pull-down menu.
- 2 Select the Update button.

Recursion Settings

The Recursion Limit panel allows the administrator to set a limit as to how many times a Lasso page may call itself using an [Include] or [Library] tag. A recursion limit will eliminate cases where a page has been programmed to recurse infinitely, and would otherwise adversely affect the operation of Lasso Service.

To set a recursion limit:

- 1 Select Yes from the Limit Recursion pull-down menu (default). Selecting No removes the recursion limit, which is not recommended.
- 2 Enter the number of times (positive integer) you want to allow a page to recurse before being stopped in the Recursion Threshold field (50 by default).
- 3 Select the Update button.

Execution Time Settings

The Lasso page Execution Time panel allows the administrator to set a time limit as to how many seconds a Lasso page (Lasso page) may run before timing out. A Lasso page execution time limit can help eliminate cases where a page has been programmed to perform endless operations that may otherwise adversely affect the operation of Lasso Service.

Important: The Lasso page execution time limit controls the amount of time Lasso waits before timing out a true Lasso operation. This does not include nor factor in any timeout thresholds present in third-party components, such as data sources, Web servers, and Web browsers.

To set a Lasso page execution time limit:

- 1 Select Yes or No from the Limit Execution Time pull-down menu to enable or disable page timeouts for the entire server (Yes by default).
- 2 Enter the number of seconds (positive integer) you want to allow a page to run before being timed out (600 by default). The format value time limit can not be set to a value below 60 (one minute).
- 3 Select Yes or No from the Allow User Override pull-down menu to enable or disable developers' abilities to set custom timeouts for individual pages using the [Lasso_ExecutionTimeLimit] tag (Yes by default).
- 4 Select the Update button.

Domain Name Lookups

This panel allows a default DNS server to be configured for this Lasso site. Lasso will normally use the system default DNS server. However, Lasso's TCP-based DNS tags are not compatible with DNS servers found in some DSL or cable modem routers which only support UDP. If Lasso returns an error when using the DNS tags then a different DNS server can be tested and configured here.

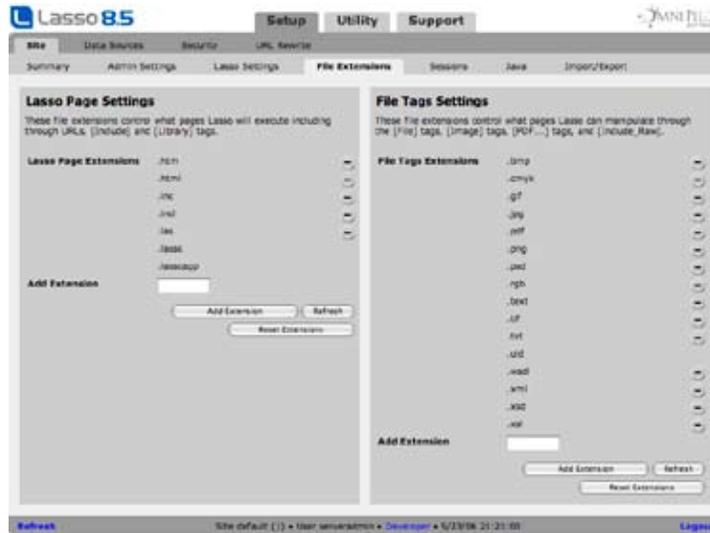
To set Lasso's default DNS server:

- 1 Enter a custom DNS server host name or IP address and select Test. Lasso will report on whether the DNS server can be accessed or not.
- 2 Select Update to accept the configured DNS server..

File Extension Settings

The File Extensions section in Site Administration allows the administrator to control what file extensions are allowed for use with Lasso. The Lasso Page Settings panel includes the extensions that can be used for Lasso pages in Lasso Professional Server. The File Tags Settings panel includes the extensions that can be used with the file tags, image tags, pdf tags, etc.

Figure 4: File Extensions Page



Adding a file extension to this page does not cause the files with that extension to be processed by Lasso, but rather grants permission for the use of files with that extension within Lasso. The difference between Lasso page settings in Site Administration and Lasso page settings in the Web server is described below.

- **Lasso Page Extension Security** – Allows the administrator to specify what Lasso pages extensions are allowed for use with Lasso. By default, the list will include .htm, .html, .inc, .incl, .las, .lasso and .LassoApp. This applies to all Lasso pages which are served via Lasso or referenced within an Lasso tag (e.g. [Include]). Files with extensions not listed here cannot be processed or included by Lasso.
- **File Tags Extension Security** – Allows the administrator to specify what files extensions are allowed for use with Lasso through the file tags, image tags, pdf tags, etc. By default, the list will include .uld, .xml, .wsdl, .text, .txt, .pdf, .jpg, .gif, .png, .bmp, .psd, .tif, .rbg, and .cmyk. This list includes file extensions that are referenced within an Lasso tag (e.g. [PDF_Doc], [File], [Include_Raw], [Image]). Files with extensions not listed here cannot be manipulated using Lasso.
- **Lasso page Settings in the Web Server** – Configuring what Lasso page extensions are actually processed by Lasso is handled in the Web Server. A file extension must be added in Site Administration and configured in the Web server before the Web server can invoke Lasso to process files. Only the .Lasso and .LassoApp extension are configured in the Web server by default after installation. See the *Extended Configuration* section of the configuration chapters in this guide for instructions on how to configure additional Lasso page extensions in the Web server.

To add a Lasso page extension in Site Administration:

- 1 Enter the extension to be allowed in the Add Extension field. If the file extension is for a Lasso page that should be processed by Lasso it needs to be entered in the Lasso page Settings panel. If the file extension is for a data file that will be manipulated by Lasso it needs to be entered in the File Tags Settings panel. All extensions must be preceded by a period (.).
- 2 Select the Add Extension button.

To remove a Lasso page extension in Site Administration:

Select the Minus (-) button next to the extension to be removed.

Note: The .Lasso, .LassoApp, and .uld extensions cannot be removed as they are required for the basic operation of Lasso. The .uld extension is used by Lasso to temporarily identify files uploaded via the Web, and is necessary if files will be uploaded via a Web interface.

Session Settings

The Sessions page allows the administrator to set site preferences for the [Sessions...] tags in Lasso Professional Server, including establishing where all session data for the site is stored.

Figure 5: Sessions Page



Session Tag Preferences

The Session Tag panel allows the administrator to set site preferences for the [Sessions...] tags. These preferences govern how sessions will propagate through HTML pages, and are described below. For information on the [Session...] tags, see the *Sessions* chapter in the Lasso 8.5 Language Guide.

Modify Tags/Attributes

The Modify Tags/Attributes field allows the administrator to specify what HTML tags and attributes the [Session...] tags will append token values to. When using sessions, these token values allow variables for a user to stay consistent from page to page. All HTML tag and attributes involved in linking a user from one page to another should be listed here. When invoked, session URL data will be added to the attributes of the tags specified, which should be set to valid paths to files. The action attribute of the <form> tag and the href attribute of the <a> tag are listed by default.

To add a new tag/attribute pair:

- 1 Next to Add Tag/Attribute, enter the name of an HTML tag in the left-most field. The < and > HTML characters should not be included as part of the tag or attribute names.
- 2 Next to Add Tag/Attribute, enter the name of the attribute of the tag to which session data will be appended in the right-most field.
- 3 Select the Add Tag/Attribute button.

To remove an existing tag/attribute pair:

Next to Modify Tag/Attributes, select the Minus (-) button next to the tag/attribute pair you wish to delete.

Avoid Prefixes

The Avoid Prefixes field allows the administrator to specify URL prefixes that should be avoided by the [Session_...] tags, meaning that no session information will be appended to links with the specified prefixes. Because session data may only be interpreted by Lasso pages, any prefix that takes a user to a page on a different server or initiates a protocol that cannot be parsed by Lasso (e.g. telnet://, file://, mailto:) should be listed here.

To add a prefix to be avoided:

- 1 Enter the URL prefix to be avoided in the Add Prefix field. Any URLs that begin with the value entered here will be avoided.
- 2 Select the Add Prefix button.

To remove a prefix:

Next to Avoid Prefixes, select the Minus (-) button next to the prefix you wish to remove.

Memory Sessions

Lasso Professional Server provides a new in-memory session driver which allows sessions to be served very fast. When the memory session driver is selected sessions will only persist until Lasso Service is restarted.

- The memory session driver can be selected by choosing Memory from the Serve Sessions From menu and clicking the Select button.
- The default SQLite storage driver can be reselected by choosing Local Default from the Serve Sessions From menu and clicking the Select button.

Multi-Server Session Support

Lasso Professional Server provides the ability to share session data with other Lasso Professional Server servers so that Lasso sessions can persist within load-balancing and redundant server configurations with multiple Lasso Professional Server servers.

Session sharing is achieved by designating a single external MySQL host to serve all Lasso session data, and then configuring each Lasso Professional Server server to store all session data on this host.

For best performance, the session tuning parameters described below should be used to establish one Lasso server as the master of the session table which will perform routine maintenance. Each of the other Lasso servers that are sharing the same MySQL table should be set to never delete expired sessions.

If multiple Lasso 8.5 (8.1 or higher) servers are sharing the same MySQL data source then the MySQL compatibility option in the session tuning parameters should be set to Lasso 8.1. This provides the best performance. If Lasso 7 and Lasso 8 servers are sharing the same MySQL data source then the MySQL Compatibility option should be set to Lasso 7/8. See the section on session tuning parameters below for more information.

Warning: Configuring Lasso to store all session data on an external MySQL host creates a dependency on that external host in order for sessions to work. It is strongly recommended that the MySQL server be on the same network as the Lasso Professional Server servers.

Configuring an External Session Host

The Session Data Server panel allows the administrator to set the MySQL host that will be used to store all session data for the current Lasso Professional Server server.

To create and use an external sessions table:

- 1 In the Serve Sessions From pull-down menu, select External Database. This specifies that a session table other than the default site.sessions table will be used.

- 2 Next to **Select Database**, select the database where to want a `sessions` table to be created. This menu contains a list of all enabled MySQL databases in the *Setup > Data Sources > Databases* section of Site Administration.
- 3 Select the **Create Sessions Table...** button. A warning will be displayed indicating that the current Lasso Professional Server server must rely on the newly created `sessions` table in the database specified for sessions to work.
- 4 Click the **OK** button. This table will be used to store all session data until the host is manually switched again.

To select and use an existing external sessions table:

- 1 In the **Server Sessions From** pull-down menu, select **External Database**. This specifies that a session table other than the default `site.sessions` table will be used.
- 2 Next to **Select Database**, select the database that contains the `sessions` table you wish to use. This menu contains a list of all enabled MySQL databases in the *Setup > Data Sources > Databases* section of Site Administration.
- 3 Select the **Select** button. A warning will be displayed indicating that the current Lasso Professional Server server must now rely on the specified `sessions` table for sessions to work.
- 4 Click the **OK** button. This table will be used to store all session data until the host is manually switched again.

Session Tuning

The session tuning preferences are advanced options that allow the behavior of the session manager to be modified for best performance in a variety of different server configurations.

- **Delete Expired at Startup** – Controls whether expired sessions are deleted automatically when the site starts up. The default is **Yes**. Setting this parameter to **No** can speed up Lasso's startup time. This parameter should be set to **No** on all all but one server if multiple servers are sharing a single MySQL table for session storage.
- **Delete Expired Frequency** – Controls how often expired sessions are deleted while Lasso is running. The session manager will never load an expired session, but periodic maintenance is required to ensure that the session table does not store a lot of expired records. The frequency can be set to one of the following values.
 - Hourly – Expired sessions are deleted about once an hour.
 - Daily – Expired sessions are deleted once a day in the hour after midnight.
 - Weekly – Expired sessions are deleted about once a week.
 - Never – Expired sessions are never deleted automatically.
 - High – Expired sessions are deleted at random about every 20th call to `[Session_Start]`. This is the default setting and the standard behavior of earlier versions of Lasso.
 - Medium – Expired sessions are deleted at random about every 200th call.
 - Low – Expired sessions are deleted at random about every 2000th call.

Setting this parameter to **Daily** or one of the other options can increase the responsiveness of Lasso since the session table never needs to be locked for maintenance during the busiest times for the site. This parameter should be set to **Never** on all all but one server if multiple servers are sharing a single MySQL table for session storage.

Note: `[Session_DeleteExpired]` can also be used in a scheduled event to delete sessions on a custom schedule.
- **Maximum Expiration Time** – Sets a maximum value for the `[Session_Start]-Expires` parameter in minutes. A setting of 0 (the default) allows any value for the `-Expires` parameter. This preference can be used to limit the expiration time of all of the sessions on the server to a reasonable value without modifying any Lasso code.

- **MySQL Compatibility** – There are two options if sessions are being stored in a shared MySQL data source. If only Lasso 8.5 (8.1 or higher) servers are sharing the storage (or if only this Lasso 8.5 server is using the storage) then the Lasso 8.1 option should be selected since this format improves the speed of the session manager. If Lasso 8 and Lasso 7 servers are sharing the storage then the Lasso 7/8 option should be selected.

Java Settings

The Java page is where the administrator can set up Lasso Professional to work with Java, so that Lasso's Java-based components may be used. This includes JDBC data source connectivity, third-party LJAPI modules, and Lasso tags which have Java-based implementations. Consult the Lasso 8 Reference for information on whether or not an Lasso tag requires Java.

The Java Settings panel allows the administrator to globally enable or disable Java-based components, and to set the Class Path and Library Path for third-party Java module class files. The Class Path and Library Path for all default Java modules installed with Lasso Professional Server is the Lasso Professional Server/LassoModules folder, which cannot be changed. Altering the Class Path and Library Path fields in the Java Settings panel will not affect how the default Java modules operate.

Figure 6: Java Page



LJAPI allows Java programmers to develop new tags and functionality in Lasso Professional Server. For more information on LJAPI, see the *Lasso Java API* chapter in the Lasso 8.5 Language Guide.

To update Java settings:

- 1 Select Enabled or Disabled from the LJAPI Modules pull-down menu to enable or disable LJAPI (Disabled by default). This includes all core Java functionality in Lasso Professional Server as well as third-party LJAPI modules.

- 2 Enter the file path to the Java class folder in the Class Path field.

Note: Multiple Class Paths or Library Paths can be specified by entering a semi-colon ; (Windows) or colon : (Mac OS X) after a path, and then entering the next full path immediately after the semi-colon or colon. There should be no spaces or carriage returns in or between any specified path(s) per JRE syntax requirements.

- 3 Enter the file path to the Java library folder in the Library Path field.

Windows Note: The default Class Path and Library Path may not be correct depending on the version of JRE installed. It is necessary to confirm the actual path to the lib folder in Windows before selecting Update.

Mac OS X Note: The Class Path and Library Path fields are unnecessary for Mac OS X installations, and are blank by default.

- 4 Select Update. A dialog box will appear informing one to double-check the Class Path and Library Path in the hard drive to ensure they are correct, and to restart Lasso Service after selecting OK.
- 5 Select OK.
- 6 Restart Lasso Service for the changes to take effect.
- 7 Return to the *Setup > Site > Java* page in Site Administration.

- 8 Select the Test button.

If the Activated field now displays Yes, then LJAPI has been successfully enabled. Otherwise, an error will be displayed informing you to double-check the Class Path and Library Path fields against the actual file path to the lib folder on the hard drive, and to make sure that Java has been installed properly. For instruction on properly installing Java, see the configuration chapters in this guide.

LJAPI is checked to ensure it is operational via an internal check for an LJAPI-based tag (e.g. [NS_Lookup]). If this tag is functioning as expected, Lasso will then change the value in the Activated field to Yes.

Import and Export Settings

The Import/Export page in Site Administration provides the ability to export the complete state of Lasso's configuration and security settings. Importing and exporting setup files is useful for backing up a particular Lasso Professional Server configuration, or setting up a new installation with the same configuration as another copy of Lasso Professional Server.

A Lasso setup file is a text file that contains all custom administration settings entered in Site Administration. The information in a Lasso setup file corresponds to the options selected in Site Administration in the Site Settings, Data Sources, and Security pages.

Care should be taken with the setup files that are exported, as these files contain all of the security settings of Lasso Professional Server. Lasso stores passwords for data sources and users in plain text. Export files which have been modified by an untrusted third-party should never be imported into Site Administration without careful review of the settings within the file.

Figure 7: Import/Export Page



Exporting Lasso Setup Files

The Export Setup panel allows the current Lasso Professional Server configuration to be exported as a text setup file to be preserved for future use.

To export a Lasso setup:

- 1 In the Export Setup panel, select the Export Setup button. This will automatically download a file that contains the contents your Lasso Professional Server setup file named Export_YYYYMMDDHHMMSS.txt.

Importing Lasso Setups

The Import Setup panel allows Lasso setup files to be imported. Only setup files exported using Lasso Professional Server may be imported.

To import a Lasso setup:

- 1 In the Import Setup panel, select the Browse button, and choose the Lasso setup file on your hard drive you wish to import.
- 2 Select the Import Setup button. A warning informs the administrator that importing a setup will overwrite all existing installation settings with the settings being imported.
- 3 Select the OK button. This will overwrite all current settings with those stored in the imported setup file.

- 4 After the import has been performed the current site is automatically stopped. Follow the Site Administration link to trigger the site to restart. You may get an error when you first click the link. Wait 10-15 seconds and try the link again.

Note: The site administrator username and password will not be overwritten.

13

Chapter 13

Setting Up Data Sources

This chapter describes how to configure data sources for use with Lasso Professional Server, and is divided into the following sections.

- *Overview* provides an overview of data source connectors in Lasso Professional Server.
- *FileMaker Pro Data Sources* provides instructions for setting up FileMaker Pro 4, 5, or 6 data sources.
- *FileMaker Server Data Sources* provides instructions for setting up FileMaker Server 7, 8, or 9 Advanced and FileMaker Server 9 data sources.
- *JDBC Data Sources* provides instructions for setting up data sources using a JDBC driver.
- *MySQL Data Sources* provides instructions for setting up a MySQL data source.
- *ODBC Data Sources* provides instructions for setting up data sources using an ODBC driver.
- *OpenBase Data Source* provides instructions for setting up OpenBase data sources.
- *Oracle Data Source* provides instructions for setting up Oracle data sources.
- *PostgreSQL Data Source* provides instructions for setting up PostgreSQL data sources.
- *Spotlight Data Source* provides instructions for configuring the Spotlight data source on Mac OS X.
- *SQL Server Data Sources* provides instructions for setting up Microsoft SQL Server data sources.
- *SQLite Data Source* describes the SQLite data source built into Lasso.
- *Other Data Sources* provides instructions for setting third-party data sources.
- *Setting Databases Preferences* describes how to set database-level preferences in Lasso Professional Server.
- *Setting Table Preferences* describes how to set table-level preferences in Lasso Professional Server, including Unicode-encoding preferences.
- *Setting Field Preferences* describes how to set field-level preferences in Lasso Professional Server.

Overview

Lasso Professional Server communicates with data sources using Lasso data source connectors, which are modular components configured using the **Setup > Data Sources** section of Site Administration. Lasso Professional Server provides built-in connectors all of the data sources listed in the following table.

Most connectors can access data sources which are installed on the same machine as Lasso or on a remote machine. Some connectors such as the Spotlight data source can only access files on the local machine. Whether or not an ODBC or JDBC data source can communicate with Lasso on a separate machine depends on whether or not the driver can communicate via TCP/IP.

Custom data source connectors for other data sources can also be created for use with Lasso Professional Server using LJAPI, LCAPI, or Lasso Tags. Information about creating and using third-party data source connectors can be found in the Language Guide.

Table 1: Lasso Professional Server 8.5 Data Sources

Data Source	Notes
FileMaker Pro	Supports FileMaker Pro 4.x, FileMaker Pro Unlimited 5.x and 6.x.
FileMaker Server	Supports FileMaker Server 7, 8, or 9 Advanced and FileMaker Server 9.
JDBC	Supports any data source with a compatible JDBC driver. See the JDBC Data Sources section for details about how to install JDBC drivers.
MySQL	Supports MySQL 3.x, 4.x, or 5.x data sources.
ODBC	Support any data source with a compatible ODBC driver. See the ODBC Data Sources section for details about how to install ODBC drivers.
OpenBase	Supports OpenBase data sources.
Oracle	Supports Oracle data sources. The Oracle "Install Client" libraries must be installed in order to activate this data source. See the Oracle Data Sources section for more information.
PostgreSQL	Supports PostgreSQL data sources. The PostgreSQL client libraries must be installed in order to activate this data source. See the PostgreSQL Data Sources section for more information.
Spotlight	Supports searching files on Mac OS X 10.4/10.5 systems only.
SQL Server	Supports Microsoft SQL Server. The SQL Server client libraries must be installed in order to activate this data source. See the SQL Server Data Sources section for more information.
SQLite	SQLite is also the internal data source which is used for the storage of Lasso's preferences and security settings.

Using Lasso Data Source Connectors

Data source connectors allow database actions to be performed via Lasso Web pages and Lasso pages. Database actions can be used in Lasso to search for records in a database that match specific criteria, to navigate through the found set from a database search, to add, update, or delete a record in a database, to fetch schema information about a database, and more. In addition, database actions can be used to execute SQL statements in SQL-compliant databases.

Interacting with data sources via Lasso Professional Server involves four steps:

- Configuring the data source application or service to accept connections from Lasso Professional Server. This is done in the data source itself, outside of Lasso Site Administration. This chapter describes configuring each data source to accept connections from Lasso Professional Server.
- Data sources can be connected to directly by specifying the host parameters directly within inlines. Using this method does not require setting up the data source in Lasso Site Administration and does not require configuring Lasso security. This method is best used when developing sites where some security can be sacrificed for coding efficiency.
- Configure Lasso Professional Server to communicate with a data source host. This involves adding the data source connection information in the *Setup > Data Sources > Hosts* section, and enabling the desired databases in the *Setup > Data Sources > Databases* section. This chapter describes creating connections with MySQL, FileMaker Pro 4, 5, or 6, FileMaker Server 7, 8, or 9 Advanced, FileMaker Server 9, and JDBC-compliant data sources.

Set up Lasso Security for the data source via the *Setup > Security* section of Site Administration. This typically involves setting database and table permissions for the AnyUser group, and then creating custom groups and users if password-protected user access is desired. This is described in the *Setting Up Security* chapter.

- Write Lasso code to interact with the data source. This is covered in the *Database Interaction* section of the Language Guide.

Viewing Lasso Data Source Connectors

The *Setup > Data Sources > Connectors* page in Site Administration is where all data source connectors can be listed, viewed, enabled, or disabled. The Connector Listing panel shows all connectors currently installed to Lasso Professional Server, the status of each connector, and links to the Hosts page for each connector.

Data sources which require additional set up will be marked as Unavailable in this listing. Consult the appropriate section of this chapter for details about how to install the additional components necessary to enable the data source. JDBC data sources will not show up unless drivers have been installed in the JDBCDrivers folder.

Figure 1: Connectors Page



Updating Data Source Connectors

Selecting a data source connector from the Connectors Listing panel will show the Connector Detail panel to the right. The Connector Detail panel displays the name of the selected data source connector, its status, and the internal name for the connector as it appears in the Lasso Service console. Data source connectors may be enabled or disabled using this panel. JDBC drivers additionally allow the specific driver type to be selected.

To update details for a selected connector:

- 1 Select Enabled or Disabled from the Status pull-down menu to enable or disable the connector.
- 2 JDBC Drivers allow the JDBC Driver Class to be selected. In general a JDBC 2.0 class should be selected if available.
- 3 Select the Update button.

FileMaker Pro Data Sources

Lasso Professional Server supports communication with FileMaker Pro 4, 5, or 6, FileMaker Server 7, 8, or 9 Advanced, or FileMaker Server 9. This section documents how to set up Lasso to communicate with the different versions of FileMaker Pro. See the next section for information about connecting to FileMaker Server.

Lasso Professional Server communicates with FileMaker Pro 4, 5, or 6 via the Web Companion, which is the built-in Web-publisher for FileMaker Pro. Using Web Companion, FileMaker Pro may be set up to run on the same machine as Lasso Professional Server or on a separate machine.

Requirements

FileMaker Pro 4.x, FileMaker Pro 5.x Unlimited, or FileMaker Pro 6.0 Unlimited for Windows or Mac OS X with the latest version of Web Companion activated and configured. FileMaker Server connectivity is documented in the next section. See the table below for all versions of FileMaker Pro's Web Companion certified to work with Lasso Professional Server.

Table 2: Supported Versions of FileMaker Pro's Web Companion

FileMaker Version	Web Companion Version
FileMaker Pro 4.x	4.0v3
FileMaker Pro Unlimited 5.0v3	5.0v6
FileMaker Pro Unlimited 5.5v2	5.5v4
FileMaker Pro Unlimited 6.0v4	6.0v2
FileMaker Server 7 Advanced	See Next Section
FileMaker Server 8 Advanced	See Next Section
FileMaker Server 9	See Next Section
FileMaker Server 9 Advanced	See Next Section

The latest updates to FileMaker Pro's products can be downloaded from FileMaker, Inc.'s Web site at:

<http://www.FileMaker.com/>

FileMaker Licensing Note: In accordance with the FileMaker Pro license agreement, only FileMaker Pro Unlimited 5 or 6 may be used with Lasso Professional Server. However, The regular version of FileMaker Pro 4, 5, or 6 may be used with Lasso Studio or Lasso Developer. For more information on the FileMaker Pro license agreement, see FileMaker's Web site at <http://www.filemaker.com>.

Configuring FileMaker Pro 4, 5, or 6

This section describes setting up FileMaker Pro 4, 5, or 6 Web Companion for use with Lasso Professional Server.

To configure FileMaker Pro Web Companion:

- 1 In FileMaker Pro, select *Edit > Preferences > Application* from the menu bar (*FileMaker Pro > Preferences > Application* in Mac OS X versions).
- 2 Select the Plug-Ins tab.
- 3 Under Enabled Plug-Ins, check the box next to Web Companion. This makes the *Configure...* button appear at the bottom of the window.
- 4 Select the *Configure...* button.
- 5 Enter the TCP/IP port through which Web Companion will communicate with Lasso Professional Server in the TCP/IP field.

Security Note: Communication with the Web Companion in FileMaker Pro 5.x and 6.x can be limited to a single IP address via checking the *Restrict access to IP address(es)* box and entering the IP address of the machine that is running Lasso Professional Server. This is highly recommended for maximum data source security.

- 6 Select OK.

Web Companion is now configured to communicate from the FileMaker Pro application via the TCP/IP port specified. Sharing via Web Companion also needs to be enabled for each FileMaker Pro database that is to be used with Lasso Professional Server.

To enable databases to be shared via Web Companion:

- 1 In FileMaker Pro, open the database to be used with Lasso Professional Server.
- 2 Select *File > Sharing* from the menu bar.
- 3 Check the box next to Web Companion.
- 4 Select OK. The FileMaker Pro database is now ready for use with Lasso Professional Server.

For proper performance, database-level security must be aligned between FileMaker Pro and Lasso Professional Server. This involves setting a master password for a FileMaker Pro database, and then entering

the password in Site Administration. Follow the procedure below for setting a master password for a FileMaker Pro database, which is then provided to Lasso Professional Server.

To set a master password for a FileMaker Pro database:

- 1 In FileMaker Pro, select *File > Access Privileges > Passwords* from the menu bar.
- 2 Under Privileges, check Access the entire file.
- 3 In the Available commands pull-down menu, select Normal.
- 4 Enter a password in the Password field.
- 5 Select Create.
- 6 Select Done.

The above steps will prevent any user from accessing a FileMaker Pro database directly without knowing the correct password. The FileMaker Pro host may now be configured in the **Setup** section of Site Administration.

FileMaker Pro Configuration Guidelines

For proper performance, adhere to the following guidelines.

- All Lasso-enabled FileMaker Pro databases must be open in FileMaker Pro before a Lasso action can be performed on the database.
- All Lasso-enabled FileMaker Pro databases must be in **Browse** mode.
- All Lasso-enabled FileMaker Pro databases may be in either single-user or multi-user mode in FileMaker Pro.
- All Lasso-enabled FileMaker Pro databases should be opened with the master password if passwords have been established, specifically so that all necessary fields, layouts, scripts, and functions (browsing, printing, exporting, creating records, etc.) are available.
- FileMaker Pro database security in Lasso Professional Server is only as secure as the Web Companion setup. It is possible for Web browsers to communicate directly with the Web Companion, which circumvents Lasso security. It is strongly recommended that database-level restrictions be enabled within FileMaker Pro, and user-level security be controlled by Lasso Professional Server security. Additionally, it is strongly recommended that only a single IP address is permitted to access Web Companion which represents the machine on which Lasso Professional Server operates.
- For tips on optimizing performance for FileMaker Pro databases, see the *FileMaker Pro Data Sources* chapter in the Lasso 8.5 Language Guide.

Creating Connections to FileMaker Pro Hosts

The **Setup > Data Sources > Hosts** page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso Professional Server communicates.

Figure 2: FileMaker Pro Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting **Enable All** will enable all hosts for the selected connector, while selecting **Disable All** will disable all data source hosts for the selected connector.

There are two FileMaker connectors in Lasso Professional Server. Lasso Connector for FileMaker Pro enables communication with FileMaker Pro versions 4, 5, or 6. Lasso Connector for FileMaker SA enables communication with FileMaker Server 7, 8, or 9 Advanced and FileMaker Server 9.

Adding Data Source Hosts

Selecting the **Add Host** button shows the **Add Host** panel to the right. The **Add Host** page allows the administrator to add a new FileMaker data source host to a selected data source connector.

To add a new FileMaker Pro host:

Make sure that the proper data source is selected for the version of FileMaker Pro that is to be accessed.

- 1 In the **Add Host** panel, enter a name for the host in the **Name** field. This will be displayed in the **Hosts Listing** panel for the host, and can be as descriptive as desired.
- 2 Enter the IP address or domain name of the host in the **IP Address/Host Name** field. This is the address of the machine running the data source application.
- 3 Enter the TCP/IP port of the data source application or service of the host in the **Port** field. See the FileMaker Pro documentation for information on where to find or set this in the data source. This is commonly 591 for the FileMaker Pro Web Companion.
- 4 Select **Enabled** from the **Status** pull-down menu to enable the host.
- 5 Enter a username for the host in the **Host Username** field. Lasso Professional Server will connect to the data source and all databases therein using this username by default. If the host does not require a username, then leave this field blank.

Note: No username should be entered for a FileMaker Pro host if **FileMaker Pro Access Privileges** is used as the security method for the FileMaker Pro data source.

- 6 Enter a password for the host in the **Host Password** field. Lasso Professional Server will connect to the data source and all databases therein using this password by default. If the host does not require a password, then leave this field blank.
- 7 Select the **Add Host** button.

Once the host is added, the new host appears in the **Hosts Listing** panel to the left. There, one can select the **List Databases** link, or select the **Databases** tab to view the databases in the specified host.

Updating and Deleting Host Settings

Selecting the name of a host shows its details in the **Host Detail** panel, which appears to the right. In the **Host Detail** panel, the administrator can update or delete the selected host by selecting the **Update** or **Delete** button.

Table Batch Change

The table batch change option allows the encoding of every table on the current host to be changed at once. Select a new encoding from the pop-up menu or select **Other...** and type a custom encoding choice. After clicking **perform every table** on the current host will be set to the specified encoding.

Note: New tables which are added to databases on the host will not be set to the selected encoding. It may be necessary to adjust the encoding of each new table after they are added or to use the batch change option after new tables are added.

Specifying FileMaker Pro Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access a FileMaker Pro data source host. See the Lasso Language Guide for full details about inline host specification.

To access a FileMaker Pro host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `fmremotedatasource`. The `-Name` should be specified as the URL of the machine hosting FileMaker Pro. The `-Port` is optional and defaults to 591. The `-Username` and `-Password` should be specified.

For example, the following code shows how a connection to a FileMaker Pro database hosted on the same machine as Lasso might appear.

```
[[Inline: -FindAll,
  -Host=(Array:
    -Datasource='fmremotedatasource',
    -Name='127.0.0.1',
    -Port=591,
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
 [Found_Count]
[/inline]
```

Enabling FileMaker Pro Databases

The last required step in configuring a data source connection is to enable the newly-connected databases in Site Administration. This provides a method for the Lasso site administrator to designate which databases can and cannot be used with Lasso.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

FileMaker Server Data Sources

Lasso Professional Server communicates with FileMaker Server 7, 8, or 9 Advanced and FileMaker Server 9 through the built-in XML interface. Lasso cannot communicate with any other products in the FileMaker 7, 8 or 9 product line such as FileMaker Pro.

Requirements

FileMaker Server 9 for Windows or Mac OS X (or FileMaker Server 9 Advanced). Or, FileMaker Server Advanced 7 or 8 for Windows or Mac OS X. The Web Publishing Engine must be installed and each database must be configured according to the instructions in the following section.

The latest versions of FileMaker Server can be downloaded from FileMaker, Inc.'s Web site at:

<http://www.FileMaker.com>

Configuring FileMaker Server 9 for Use With Lasso

This section describes setting up FileMaker Server 9 or FileMaker Server 9 Advanced for use with Lasso Professional Server.

To configure FileMaker Server 9:

Follow the instructions included with FileMaker Server 9 carefully. Setting up FileMaker Server 9 is considerably easier than setting up earlier versions of FileMaker Server.

- Make sure that the Web Serving options are turned on and that the XML interface is enabled.
- The databases which are to be accessed by Lasso must be in the FileMaker Server *Data/Databases* folder and must be *Open* within FileMaker Server.
- Each database to be accessed by Lasso must have the *fmxml* keyword added to the Extended Privileges section of the Accounts & Privileges dialog box. The username and password which are entered into Lasso Administration must use a Privilege Set which has access to this extended privilege.
- FileMaker Pro database security in Lasso Professional Server is only as secure as the Publishing Engine setup. It is possible for Web browsers to communicate directly with the Publishing Engine, which circumvents Lasso security. It is strongly recommended that the security features of FileMaker Pro be used to secure accessible databases.
- It is strongly recommended that only a single IP address is permitted to access the Publishing Engine which represents the machine on which Lasso Professional Server operates.
- For tips on optimizing performance for FileMaker databases, see *Chapter 10: FileMaker Data Sources* in the Lasso Language Guide.

Configuring FileMaker Server 7 or 8 Advanced for Use With Lasso

This section describes setting up FileMaker Server Advanced for use with Lasso Professional Server.

To configure FileMaker Server Advanced:

Follow the instructions included with FileMaker Server Advanced carefully. There are several steps in the process which are not obvious and require reading the document to set up properly. Configuring FileMaker Server Advanced is beyond the scope of this documentation, but some common pitfalls are listed below.

- Make sure both FileMaker Server and the FileMaker Server Advanced Publishing Engine are installed. The machine with the Publishing Engine must be running a supported Web server.
- Configure FileMaker Server with a Client Services identifier and passcode. Enter this same identifier and passcode in the Web Publishing Administration Console.
- Ensure XML Publishing is turned on in the Web Publishing Administration Console.
- The databases which are to be accessed by Lasso must be in the FileMaker Server *Data/Databases* folder and must be *Open* within FileMaker Server.
- Each database to be accessed by Lasso must have the *fmxml* keyword added to the Extended Privileges section of the Accounts & Privileges dialog box. The username and password which are entered into Lasso Administration must use a Privilege Set which has access to this extended privilege.
- FileMaker Pro database security in Lasso Professional Server is only as secure as the Publishing Engine setup. It is possible for Web browsers to communicate directly with the Publishing Engine, which circumvents Lasso security. It is strongly recommended that the security features of FileMaker Pro be used to secure accessible databases.
- It is strongly recommended that only a single IP address is permitted to access the Publishing Engine which represents the machine on which Lasso Professional Server operates.
- For tips on optimizing performance for FileMaker databases, see *Chapter 10: FileMaker Data Sources* in the Lasso Language Guide.

Creating Connections to FileMaker Server Hosts

The *Setup > Data Sources > Hosts* page in Lasso Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso Professional Server communicates.

Figure 3: FileMaker Server Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector.

Adding Data Source Hosts

Selecting the Add Host button shows the Add Host panel to the right. The Add Host page allows the administrator to add a new FileMaker data source host to a selected data source connector.

To add a new FileMaker Server host:

- 1 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.
- 2 Enter the IP address or domain name of the host in the IP Address/Host Name field. This is the address of the machine running the data source application.
- 3 Enter the TCP/IP port of the data source application or service of the host in the Port field. See the FileMaker Server documentation for information on where to find or set this in the data source. This is commonly 80 for FileMaker Server.
- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username for the host in the Host Username field. Lasso Professional Server will connect to the data source and all databases therein using this username by default. If the host does not require a username, then leave this field blank.
- 6 Enter a password for the host in the Host Password field. Lasso Professional Server will connect to the data source and all databases therein using this password by default. If the host does not require a password, then leave this field blank.
- 7 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host.

Updating and Deleting Host Settings

Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

Specifying FileMaker Server Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access a FileMaker Server data source host. See the Lasso Language Guide for full details about inline host specification.

To access a FileMaker Server host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `fmserveradvanceddatasource`. The `-Name` should be specified as the URL of the machine hosting FileMaker Server. The `-Port` is optional and defaults to 80. The `-Username` and `-Password` should be specified.

For example, the following code shows how a connection to a FileMaker Server data source hosted on the same machine as Lasso might appear.

```
[[Inline: -FindAll,
  -Host=(Array:
    -Datasource='fmserveradvanceddatasource',
    -Name='127.0.0.1',
    -Port=80,
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
  [Found_Count]
[/inline]
```

Enabling FileMaker Server Databases

The last required step in configuring a data source connection is to enable the newly-connected databases in Site Administration. This provides a method for the Lasso site administrator to designate which databases can and cannot be used with Lasso.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

JDBC Data Sources

Lasso Professional Server communicates with JDBC data sources via JDBC drivers, which are made available for a data source either by the manufacturer or by a third party. In order for a data source to be JDBC-compliant, JDBC connectivity must be featured in the data source, and a JDBC driver must be available. Popular JDBC-compliant data sources include Microsoft SQL Server, Sybase, DB2, and Frontbase, as well as many others. For more information on JDBC connectivity for a particular data source, see the data source documentation or contact the data source manufacturer.

Lasso Professional Server functions as its own JDBC driver manager. Once JDBC connectivity is installed or is available for a data source, the JDBC driver is loaded in Lasso Professional Server to allow Lasso to connect. Lasso Professional Server uses the JDBC 3.0 industry-standard API, and can connect to any databases using drivers that are certified as JDBC 1.0, JDBC 2.0, or JDBC 3.0 compatible.

JRE Version Note: Lasso Professional Server can use JDBC drivers that are version 2.0 or higher provided that the JRE installed to the operating system also supports them. Java 1.4.x standard installations or greater on all operating systems instantly support JDBC 2.0.

Requirements

The following requirements must be fulfilled in order for a data source to communicate with Lasso Professional Server via JDBC:

- Installation of JDBC-compatible data source with any required patches or components for JDBC connectivity to a machine accessible by Lasso Professional Server.
- JDBC driver compatible with data source.
- Documentation for JDBC driver. This should contain information on what kind of connection URL information is required for Lasso Professional Server to connect via JDBC. This information varies from data source to data source.
- Installation of JRE 1.4 on the machine running Lasso Professional Server (Windows only). See the *Configuring on Windows* chapter for more information.

Setting Up JDBC Data Sources

The following procedure describes setting up a generic JDBC data source for use with Lasso Professional Server. These procedures may vary depending on the data source, and not every data source will require additional component installations to use JDBC. The documentation for the JDBC driver is important in determining what connection information is needed to allow Lasso to connect to the JDBC-compliant data source using that JDBC driver.

Important: To ease the process of setting up specific JDBC data sources, LassoSoft Support Central articles are available that provide instructions on how to set up specific JDBC data sources to communicate with Lasso Professional Server. These articles are available at <http://support.lassosoft.com>.

To set up a data source to communicate with Lasso Professional Server via JDBC:

- 1 Install any patches or components required for JDBC connectivity to the data source. See the data source documentation or contact the data source manufacturer for more information.
- 2 Start the data source application or service. The data source must be running to be accessed by Lasso Professional Server.
- 3 Lasso Professional Server will require access privileges to the data source for any or all levels of security (database-level, table-level, field-level, etc.). This can be achieved by creating a new user in the data source for use by Lasso, or by allowing Lasso to use an existing user account for data access. This user information will be used in the *Setup > Data Sources > Hosts* section of Site Administration.
- 4 Install all required JDBC drivers for the data source to the `JDBCDrivers` folder in the Lasso Professional Server application folder on the machine running Lasso Service. Any support libraries required by the data source can be placed in the `Javalibraries` folder.
- 5 Restart Lasso Service. During startup, Lasso Service scans the `JDBCDrivers` folder and lists all valid JDBC drivers in the *Setup > Data Sources > Connectors* page of Site Administration. Lasso Professional Server acts as its own JDBC driver manager, and each JDBC driver represents a data source connector.
- 6 Select the desired class from the connector using the *Setup > Data Sources > Connectors* section of Site Administration. Lasso will select the most compatible class by default.
- 6 Create a connection to a JDBC data source host using the *Setup > Data Sources > Hosts* section of Site Administration. This will involve entering a connection URL that contains the host name and other information required by the JDBC driver. The JDBC driver documentation should specify how the connection URL should be formatted.

Data Source Hosts

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected JDBC data source connector, and allows the administrator to add new hosts. The Host Listing panel shows a listing of all the hosts available for the current connector.

Figure 5: JDBC Hosts Page



Unique Add Host and Host Detail panels will be displayed depending on the JDBC data source. These panels allow the administrator to enter additional information that may be required to connect to specific JDBC data sources via specific JDBC drivers.

Both the Add Host and Host Detail panels display the following fields for most JDBC data source hosts:

- **Name** – This is the name of the host that will be displayed in the Hosts Listing panel. The properties of this field are the same as for other hosts.
- **Connection URL** – The connection URL contains the host name or IP address and TCP/IP port of the JDBC data source host in addition to any URL prefixes required by the JDBC data source. Instructions for what the connection URL (or connection string) should look like for a particular JDBC data source should be documented in the JDBC driver documentation. A common connection URL format is `jdbc:company:product://address:port`.
- **Status** – Specifies whether the host is Enabled or Disabled in Lasso Professional Server. The properties of this field are the same as for other hosts.
- **Host Username** – The username for the host. The properties of this field are the same as for other hosts.
- **Host Password** – The password for the host. The properties of this field are the same as for other hosts.
- **Host Schema** – The schema for the host, which specifies what tables and fields will be available for the databases in the host. This pull-down menu applies only to JDBC data sources that support named schema ownership (e.g. Frontbase, Sybase). If your data source does not support named schema ownership, then only a value of (None) will be shown.

Additional Fields Note: Additional host property fields may appear depending on the JDBC driver. Consult the JDBC driver documentation for instructions on how to use these additional property fields if they appear.

A JDBC data source host may be added or updated by entering the information above in the Add Host or Host Detail panel, and selecting the Add Host or Update Host button, respectively.

Configuration Tip: See the LassoSoft Support Central Archives at <http://support.lassosoft.com> for specific configuration instructions for JDBC data sources that are known to work with Lasso Professional Server.

Enabling JDBC Databases

The last required step in configuring a data source connection is to enable the newly-connected databases in Site Administration. This provides a method for the Lasso site administrator to designate which databases can and cannot be used with Lasso.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

MySQL Data Sources

This section provides instructions for setting up a MySQL data source for use with Lasso Professional Server via Lasso Connector for MySQL. A MySQL service running on a host machine is able to communicate with Lasso Professional Server via a TCP/IP port. This can be on the same machine as Lasso Professional Server, or on a separate machine. For more information on MySQL, see the following URL:

<http://www.mysql.com>

Requirements

Lasso Professional Server is certified to work with default installations of MySQL 3.23, MySQL 4.x, and MySQL 5.x (release versions). Newer versions or modified versions of MySQL may work with Lasso Professional Server, but are not officially supported.

For Lasso Professional Server to successfully communicate with a MySQL host, the following requirements must be met.

- The MySQL service must be running.
- MySQL must be assigned a port with no conflicts in which to communicate. For standard versions of MySQL, this is port 3306 by default.
- Access privileges must be assigned in MySQL to the machine running Lasso Professional Server for any or all levels (database-level, table-level, column-level, etc.) of security.

Note: Links to third party distributions and tools are provided for your convenience and were accurate when this manual was written. LassoSoft cannot be responsible for the availability or suitability of software downloaded from third party Web sites.

Configuring MySQL Hosts For Use With Lasso

MySQL is operated via a command-line interface application, which is normally located in the bin directory of the MySQL installation on the client machine. For information on how to use this, consult the MySQL documentation.

Security for MySQL data sources can be set at any level (MySQL-level, database-level, table-level, etc.). All permissions for all levels of security need to be given to Lasso Professional Server for unrestricted operation. This involves setting a new user and password for Lasso Professional Server in MySQL, then entering the username and password in Site Administration. Follow the procedure below for granting all permissions to Lasso Professional Server in MySQL using the MySQL command line utility.

To grant all permissions to Lasso Professional Server in MySQL:

- 1 In the command line utility, log in to MySQL as the root user by entering the following command:

```
mysql -u root -p your_password;
```

Replace `your_password` with the root password for MySQL specified during the MySQL installation. For instructions on how to set or change the MySQL root password, see the MySQL documentation.

- 2 Enter the following to assign a username and password for Lasso Professional Server to gain access to all levels of security in MySQL.

```
GRANT ALL ON *.* TO Username@Hostname IDENTIFIED BY "Password";
```

Replace Username and Password with the username and password values you wish to assign to Lasso Professional Server. These will be used in the *Setup > Data Sources > Databases* section in Site Administration, as described in the *Setting Up Data Sources* chapter. Replace Hostname with the host name or IP address of the machine on which Lasso Professional Server is installed. This will restrict the access of this username and password to the machine running Lasso Professional Server for added security. ALL represents all MySQL action permissions (e.g. INSERT, SELECT, DELETE, etc.), and the *.* characters represent all levels of security in MySQL (e.g. database-level, table-level, etc.).

Lasso Professional Server now has all appropriate permissions to communicate with MySQL, and the MySQL host may now be configured in the Setup section of Site Administration. See the *Setting Up Data Sources* chapter for more information.

Note: It is possible to assign limited privileges to Lasso Professional Server one at a time by replacing ALL with an individual permission (e.g. INSERT, SELECT, DELETE), and replacing *.* with a specific database or table name. This will restrict the functionality of Lasso Professional Server to the privileges that are assigned to it. For example, giving Lasso Professional Server only the SELECT privilege will allow a MySQL database to be searched using Lasso, but records cannot be added, updated, or deleted using Lasso.

Creating Connections to MySQL Hosts

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso Professional Server communicates.

Figure 6: MySQL Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector.

Adding MySQL Hosts

Selecting the Add Host button shows the Add Host panel to the right. The Add Host page allows the administrator to add a new MySQL data source host to a selected data source connector.

To add a new MySQL host:

- 1 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.
- 2 Enter the IP address or domain name of the host in the IP Address/Host Name field. This is the address of the machine running the data source application.
- 3 Enter the TCP/IP port of the data source application or service of the host in the Port field. This is commonly 3306 for MySQL data sources.
- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username for the host in the Host Username field. Lasso Professional Server will connect to the data source and all databases therein using this username.

- 6 Enter a password for the host in the Host Password field. Lasso Professional Server will connect to the data source and all databases therein using this password.
- 7 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host.

MySQL 4.1/5.x Character Sets

The option to use MySQL 4.1/5.x character sets allows Lasso to insert data into MySQL tables using the native encoding specified when the table is constructed. This option should only be used on MySQL 4.1/5.x data sources.

When this option is on the encoding for every table in the MySQL host should be set to UTF-8 for highest fidelity. Lasso will transmit UTF-8 to the remote host and then instruct MySQL to convert that data to the native encoding appropriate for the destination table.

Note: Versions of Lasso prior to 8.1 always connected to MySQL using an ISO-8859-1 character set. Data stored in this way can be retrieved by Lasso, but cannot be retrieved by other MySQL clients without performing a character set translation. The Use MySQL 4.1/5.x Character Sets option should be left off until the data within the existing tables can be fixed.

Updating and Deleting Host Settings

Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

Specifying MySQL Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access a MySQL data source host. See the Lasso Language Guide for full details about inline host specification.

To access a MySQL host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `mysqls`. The `-Name` should be specified as the URL of the machine hosting MySQL. The `-Port` is optional and defaults to 3306. The `-Username` and `-Password` should be specified.

For example, the following code shows how a connection to a MySQL data source hosted on the same machine as Lasso might appear.

```
[Inline: -FindAll,
  -Host=(Array:
    -Datasource='mysqls',
    -Name='127.0.0.1',
    -Port=3306,
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
[Found_Count]
[/inline]
```

Enabling MySQL Databases

The last required step in configuring a data source connection is to enable the newly-connected databases in Site Administration. This provides a method for the Lasso site administrator to designate which databases can and cannot be used with Lasso.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

ODBC Data Sources

This section provides instructions for setting up data source through ODBC for use with Lasso Professional Server. An ODBC driver set up as a system data source (System DSN) is required in order to access the data source through Lasso. The data source can be on the same machine as Lasso Professional Server or on a separate machine. For more information on ODBC, see the documentation included with your operating system:

Requirements

Lasso's ODBC data source requires a compatible ODBC driver which has been configured as a System DSN in the ODBC control panel.

- **Windows** – ODBC data sources are configured using the ODBC Data Source Administrator which is normally accessed through *Settings > Control Panels > Administrative Tools > Data Sources (ODBC)*. Lasso can access data sources configured as System DSNs.
- **Mac OS X** – ODBC data sources are configured using the ODBC Administrator which is can be found in the Applications/Utilities folder. Lasso can access data sources configured as System DSNs.
- **Linux** – Consult the documentation of the ODBC drivers for information about how to set up data sources on Linux. Many ODBC drivers ship with a control panel which allows configuration of those drivers. Lasso can access data sources configured as System DSNs.

Configuring ODBC Hosts For Use With Lasso

Consult the documentation for your data sources and ODBC drivers for details about how to secure access to the data made available through the driver. Most data sources will require the following steps:

- 1 Install your ODBC driver using the provided installer or instructions.
- 2 Create a System DSN in the ODBC administration application (see above). Note that the System DSN name, username, and password which are configured here will need to be entered in Lasso.
- 3 Locate and configure the SQL.ini file for your driver. This file sets the options for your ODBC driver including the location of your data source. Consult your driver's documentation for details about where to find this file and what options can be configured.
- 4 Follow the steps below to add the data source to Lasso.

Adding ODBC Hosts to Lasso

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso communicates and has an associated username and password.

Figure 7: ODBC Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector. Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

To add a new ODBC host:

- 1 Select the Add Host button to show the Add Host panel to the right.
- 2 In the Add Host panel, enter the name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired. Often this will be the same as the System DSN.
- 3 Enter the System DNS of the in the the Data Source Name field. For example, a connection to the System DSN ExampleDatabase might appear as follows..

ExampleDatabase

- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username and password for the host in the Host Username and Password fields. Lasso Professional Server will connect to the data source and all databases therein using this username and password.
- 6 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host. Any databases which will be accessed through Lasso need to be enabled.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

Specifying ODBC Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access an ODBC data source host. See the Lasso Language Guide for full details about inline host specification.

To access an ODBC host directly in an inline:

The -Host parameter can be used to specify all of the connection parameters for a data source host. The -Host array should contain the following elements. The -Datasource should be specified as `odbc`. The -Name should be specified as the System DSN, e.g. `exampledatabase`. The -Username and -Password should be specified if required by the ODBC driver.

For example, the following code shows how a connection to the same ODBC data source which was configured above might appear.

```
[Inline: -FindAll,
-Host=(Array:
  -Datasource='odbc',
  -Name='exempldatabase',
  -Username='lassouser',
  -Password='mysecretpassword'),
-Database='mydatabase',
-Table='example']
[Found_Count]
[/inline]
```

OpenBase Data Sources

This section provides instructions for setting up an OpenBase data source for use with Lasso Professional Server. An OpenBase service running on a host machine is able to communicate with Lasso Professional Server via a TCP/IP port. The data source can be on the same machine as Lasso Professional Server or on a separate machine. For more information on OpenBase, see the following URL:

<http://www.openbase.com/>

Requirements

Lasso Professional Server has been tested with the following versions of this data source.

- OpenBase SQL 9.1

Configuring OpenBase Hosts For Use With Lasso

The OpenBase database server must be configured with a user which has access to all of the databases, tables, and other resources which will be published through Lasso. Consult the OpenBase documentation for help configuring OpenBase's built-in security.

Adding OpenBase Hosts to Lasso

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso communicates and has an associated username and password.

Figure 8: OpenBase Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector. Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

To add a new OpenBase host:

- 1 Select the Add Host button to show the Add Host panel to the right.
- 2 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.
- 3 Enter the IP address or domain name of the host and the database name in the Connection URL field. For example, a connection to the machine `openbase.example.com` to access `mydatabase` might appear as follows..

```
openbase.exampe.com/mydatabase
```

The Additional Properties field can be left blank.

- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username and password for the host in the Host Username and Password fields. Lasso Professional Server will connect to the data source and all databases therein using this username and password.
- 6 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host. Any databases which will be accessed through Lasso need to be enabled.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

Specifying OpenBase Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access an OpenBase data source host. See the Lasso Language Guide for full details about inline host specification.

To access an OpenBase host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `openbaseds`. The `-Name` should be specified the same as the Connection URL described above, e.g. `openbase.exampe.com/mydatabase`. The `-Username` and `-Password` must be specified.

For example, the following code shows how a connection to the same OpenBase database server which was configured above might appear.

```
[Inline: -FindAll,
  -Host=(Array:
    -Datasource='openbaseds',
    -Name='openbase.example.com/mydatabase',
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
[Found_Count]
[/inline]
```

Oracle Data Sources

This section provides instructions for setting up an Oracle data source for use with Lasso Professional Server. An Oracle service running on a host machine is able to communicate with Lasso Professional Server via a TCP/IP port. The data source can be on the same machine as Lasso Professional Server or on a separate machine. For more information on Oracle, see the following URL:

<http://www.oracle.com/>

Requirements

Lasso Professional Server has been tested with the following versions of this data source.

- Oracle Database 10g

Note: A universal build of the Oracle client library for Intel-based Macintosh installations was not available when Lasso 8.5 was released. The Oracle data source will not function on these machines until Oracle releases a universal version of the instant client.

Lasso's Oracle data source requires the Oracle "Instant Client" to be installed in order to access Oracle hosted on another machine. If Lasso is hosted on the same machine as Lasso then this is not necessary. The instant client can be downloaded from the following Web site. Make sure to download just the basic instant client files rather than the complete Oracle 10g client or database installation.

<http://download.oracle.com/>

- **Windows** – Decompress the "Instant Client" download and copy all the .dll files within directly into the Lasso Professional 8 application folder. Restart Lasso Service in order to access the Oracle data source.
- **Mac OS X** – Decompress the instant client. Download the following installer and copy the included .pkg file into the "Instant Client" folder (i.e. instantclient10_1). Run the installer by double clicking on the .pkg file. This installer copies the libraries provided by Oracle into their proper locations within the OS.

<http://www.kyngchaos.com/files/software/unixport/OCIinstall.zip>

After running the installer open a Terminal window and copy and paste the following commands. The sudo command will require that you enter your password in order to continue. This command creates links so that Lasso can find the Oracle libraries at the location where the installer placed them.

```
sudo ln -sf /usr/local/oracle/lib/* /usr/local/lib/
```

- **Linux** – Decompress the instant client and follow the instructions on the Web site to place the files within the library search path for Lasso.

Note: Links to third party distributions and tools are provided for your convenience and were accurate when this manual was written. LassoSoft cannot be responsible for the availability or suitability of software downloaded from third party Web sites.

Configuring Oracle Hosts For Use With Lasso

The Oracle database server must be configured with a user which has access to all of the databases, tables, and other resources which will be published through Lasso. Consult the Oracle documentation for help configuring Oracle's built-in security. The Oracle Web site has a Getting Started section which explains how to install and perform basic configuration of an Oracle database server.

<http://www.oracle.com/technology/getting-started/index.html>

Adding Oracle Hosts to Lasso

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso communicates and has an associated username and password.

Important: If the client libraries for Oracle have not been installed properly then a warning message will appear on this page. Consult the *Requirements* section above for instructions to install the Oracle client libraries which Lasso requires.

Figure 9: Oracle Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector. Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

To add a new Oracle host:

- 1 Select the Add Host button to show the Add Host panel to the right.
- 2 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.
- 3 Enter two slashes, the IP address or domain name of the host, the port (defaults to 1521), and the name of the database which is to be accessed in the Connection URL field. For example, a connection to the machine oracle.example.com on the standard port to access a database named mydatabase might appear as follows.

```
//oracle.example.com:1521/mydatabase
```

The Additional Properties field can be left blank.

- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username and password for the host in the Host Username and Password fields. Lasso Professional Server will connect to the data source and all databases therein using this username and password.
- 6 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host. Any databases which will be accessed through Lasso need to be enabled.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

Specifying Oracle Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access an Oracle data source host. See the Lasso Language Guide for full details about inline host specification.

To access an Oracle host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `oracle`. The `-Name` should be specified the same as the Connection URL described above, e.g. `oracle.example.com:1521/mydatabase`. The `-Username` and `-Password` must be specified.

For example, the following code shows how a connection to the same Oracle database server which was configured above might appear.

```
[[Inline: -FindAll,
  -Host=(Array:
    -Datasource='oracle',
    -Name='//oracle.example.com:1521/mydatabase',
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
 [Found_Count]
[/inline]
```

PostgreSQL Data Sources

This section provides instructions for setting up a PostgreSQL data source for use with Lasso Professional Server. A PostgreSQL service running on a host machine is able to communicate with Lasso Professional Server via a TCP/IP port. The data source can be on the same machine as Lasso Professional Server or on a separate machine. For more information on PostgreSQL, see the following URL:

<http://www.postgresql.org/>

Requirements

Lasso Professional Server has been tested with the following versions of this data source.

- PostgreSQL 8.x

On any platform it is necessary to install PostgreSQL on the Lasso machine, even in order to access a copy of PostgreSQL hosted on another machine. Distributions of PostgreSQL for Windows and Linux can be found at the PostgreSQL Web site.

<http://www.postgresql.org/>

- **Windows** – this URL to download the latest version of the ODBC driver for PostgreSQL. This download includes binary versions of the libraries which Lasso requires in order to access PostgreSQL.

<http://www.postgresql.org/ftp/odbc/versions/dll/>

Decompress the zip archive and copy all of the included `.dll` files into the Lasso Professional 8 application folder and restart Lasso Service.

- **Mac OS X** – Mac OS X binaries can be found at one of the following Web sites. The latter includes a universal build of PostgreSQL for Intel-based Macintosh computers.

<http://www.entropy.ch/software/macosx/postgresql/>

<http://www.kyngchaos.com/software/unixport/>

After running the installer open a Terminal window and copy and paste the following command. The `sudo` command will require that you enter your password in order to continue. This command creates links so that Lasso can find the PostgreSQL libraries at the location where the installer placed them.

```
sudo ln -s /usr/local/pgsql/lib/libpq.dylib .
```

- **Linux** – Download the PostgreSQL database and build the client libraries. Follow the instructions on the Web site to place the files within the library search path for Lasso.

Note: Links to third party distributions and tools are provided for your convenience and were accurate when this manual was written. LassoSoft cannot be responsible for the availability or suitability of software downloaded from third party Web sites.

Configuring PostgreSQL Hosts For Use With Lasso

The PostgreSQL database server must be configured with a user which has access to all of the databases, tables, and other resources which will be published through Lasso. Consult the PostgreSQL documentation for help configuring PostgreSQL's built-in security.

<http://www.postgresql.org/docs/manuals/>

Adding PostgreSQL Hosts to Lasso

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso communicates and has an associated username and password.

Important: If the client libraries for PostgreSQL have not been installed properly then a warning message will appear on this page. Consult the *Requirements* section above for instructions to install the PostgreSQL client libraries which Lasso requires.

Figure 10: PostgreSQL Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector. Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

To add a new PostgreSQL host:

- 1 Select the Add Host button to show the Add Host panel to the right.
- 2 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.

- 3 Enter the IP address or domain name of the host. For example, a connection to the machine `postgresql.example.com` might appear as follows..

```
postgresql.example.com
```

The Additional Properties field can be left blank.

- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username and password for the host in the Host Username and Password fields. Lasso Professional Server will connect to the data source and all databases therein using this username and password.
- 6 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host. Any databases which will be accessed through Lasso need to be enabled.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

Specifying PostgreSQL Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access an PostgreSQL data source host. See the Lasso Language Guide for full details about inline host specification.

To access a PostgreSQL host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `postgresql`. The `-Name` should be specified the same as the Connection URL described above, e.g. `postgresql.example.com`. The `-Username` and `-Password` must be specified.

For example, the following code shows how a connection to the same PostgreSQL database server which was configured above might appear.

```
[Inline: -FindAll,
  -Host=(Array:
    -Datasource='postgresql',
    -Name='postgresql.example.com',
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
[Found_Count]
[/inline]
```

Spotlight Data Source

The Spotlight data source is used to search the file system on machines with Mac OS X 10.4/10.5. This data source is not supported on earlier versions of Mac OS X or on Windows or Linux.

The Spotlight data source has one pre-defined host and several databases which represent different metadata sets that can be searched. The Lasso Site Administrator must enable the metadata sets which they want to make available for searching.

Consult the *Other Data Sources* chapter of the Lasso Language Guide for full details about how to perform searches using the Spotlight data source.

Apple's developer information page about Spotlight provides a conceptual background to how Spotlight works, what types of metadata are indexed, and how to run Spotlight queries from the command line.

<http://developer.apple.com/macosx/spotlight.html>

Requirements

The Spotlight data source requires Mac OS X 10.4/10.5. The Spotlight data source can only be used to search for files and folders which have been indexed by the Spotlight indexer.

Sites will only be able to return Spotlight search results for files contained within their Web server root unless the Server Administrator has granted the site permission to access files outside the root and appropriate file permissions have been granted to the user who is performing the search.

Enabling Spotlight Metadata Sets (Databases)

Lasso defines four Spotlight metadata sets as databases within Lasso Site Administration. A set will also be defined for each volume which is mounted on the machine hosting Lasso Service. Each set must be enabled and appropriate permissions assigned to it before they can be searched by the site user. Some of the sets also require file permissions to be configured. Each metadata set can be enabled by visiting the *Site > Data Sources > Databases* section of Lasso Site Administration.

- **Spotlight_Site** – This set corresponds to the Web site root. Once this “database” is enabled it can be searched by any Lasso user who has permission to access it. No additional file permissions are required.
- **Spotlight_User** – This set corresponds to the Lasso Professional 8 application folder. The site must be assigned permission to access files outside of the Web server root and the Lasso user must have permission to access those files in order to use this set.
- **Spotlight_Computer** – This set corresponds to the root of the machine hosting Lasso Service. The site must be assigned permission to access files outside of the Web server root and the Lasso user must have permission to access those files in order to use this set. Note that the search results may contain files which the Lasso user is not capable of manipulating through the file tags.
- **Spotlight_Network** – This set corresponds to the local area network of the machine hosting Lasso Service. The site must be assigned permission to access files outside of the Web server root and the Lasso user must have permission to access those files in order to use this set.
- **/Volumes/MacintoshHD** – Each local hard disk, mounted CD or DVD, mounted disk image, and file server will appear as a separate “database” with the name */Volumes/* followed by the volume name.

SQL Server Data Sources

This section provides instructions for setting up a Microsoft SQL Server data source for use with Lasso Professional Server. A Microsoft SQL Server service running on a Windows host machine is able to communicate with Lasso Professional Server via a TCP/IP port. The data source can be on the same machine as Lasso Professional Server or on a separate machine. For more information on Microsoft SQL Server, see the following URL:

<http://www.microsoft.com/sql/>

Requirements

Lasso Professional Server has been tested with the following versions of this data source.

- Microsoft SQL Server 2005

On Mac OS X and Linux it is necessary to install a set of client libraries in order to access Microsoft SQL Server.

- **Windows** – Windows machines should already have the necessary client libraries for SQL Server installed.
- **Mac OS X** – Install the FreeTDS libraries in order to access Microsoft SQL Server. The data source will show up as unavailable unless these libraries are installed. The Mac OS X libraries can be found at HMUG. Follow the instructions on the Web page to install the libraries. The Mac OS X developer tools are required and can be found on the Mac OS X install DVD or downloaded direct from Apple.

http://www.hmug.org/pub/MacOS_X/BSD/Libraries/System/freetds/

- **Linux** – Install the FreeTDS libraries in order to access Microsoft SQL Server. The data source will show up as unavailable unless these libraries are installed. The Linux libraries are available directly from FreeTDS.

<http://www.freetds.org/>

Note: Links to third party distributions and tools are provided for your convenience and were accurate when this manual was written. LassoSoft cannot be responsible for the availability or suitability of software downloaded from third party Web sites.

Configuring SQL Server Hosts For Use With Lasso

The SQL Server database server must be configured with a user which has access to all of the databases, tables, and other resources which will be published through Lasso. Consult the SQL Server documentation for help configuring SQL Server's built-in security.

<http://www.microsoft.com/sql/>

Adding SQL Server Hosts to Lasso

The *Setup > Data Sources > Hosts* page in Site Administration displays all data source hosts for a selected data source connector, and allows the administrator to add new hosts. A host is a computer that serves the databases with which Lasso communicates and has an associated username and password.

Important: If the client libraries for SQL Server have not been installed properly then a warning message will appear on this page. Consult the *Requirements* section above for instructions to install the SQL Server client libraries which Lasso requires.

Figure 11: SQL Server Hosts Page



The Host Listing panel shows a listing of all the hosts available for the current connector. Selecting Enable All will enable all hosts for the selected connector, while selecting Disable All will disable all data source hosts for the selected connector. Selecting the name of a host shows its details in the Host Detail panel, which appears to the right. In the Host Detail panel, the administrator can update or delete the selected host by selecting the Update or Delete button.

To add a new SQL Server host:

- 1 Select the Add Host button to show the Add Host panel to the right.

- 2 In the Add Host panel, enter a name for the host in the Name field. This will be displayed in the Hosts Listing panel for the host, and can be as descriptive as desired.
- 3 Enter the IP address or domain name of the host. For example, a connection to the machine `sqlserver.example.com` for the database `mydatabase` might appear as follows..

```
sqlserver.example.com\mydatabase
```

The Additional Properties field can be left blank.

- 4 Select Enabled from the Status pull-down menu to enable the host.
- 5 Enter a username and password for the host in the Host Username and Password fields. Lasso Professional Server will connect to the data source and all databases therein using this username and password.
- 6 Select the Add Host button.

Once the host is added, the new host appears in the Hosts Listing panel to the left. There, one can select the List Databases link, or select the Databases tab to view the databases in the specified host. Any databases which will be accessed through Lasso need to be enabled.

The section on Database Preferences which follows the individual data source sections includes instructions for enabling database and modifying database preferences. For information on setting table, and field preferences, see the corresponding sections at the end of this chapter. For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter.

Specifying SQL Server Hosts in Inlines

Setting up a data source host and properly configuring Lasso security is the best way to ensure that access to the data source is fully controlled. However, it can sometimes be beneficial to access a data source host without a lot of configuration. This section describes how to construct an inline which access an PostgreSQL data source host. See the Lasso Language Guide for full details about inline host specification.

To access a SQL Server host directly in an inline:

The `-Host` parameter can be used to specify all of the connection parameters for a data source host. The `-Host` array should contain the following elements. The `-Datasource` should be specified as `sqlserver`. The `-Name` should be specified the same as the Connection URL described above, e.g. `sqlserver.example.com\mydatabase`. The `-Username` and `-Password` must be specified.

For example, the following code shows how a connection to the same SQL Server database server which was configured above might appear.

```
[Inline: -FindAll,
  -Host=(Array:
    -Datasource='sqlserver',
    -Name='sqlserver.example.com\mydatabase',
    -Username='lassouser',
    -Password='mysecretpassword'),
  -Database='mydatabase',
  -Table='example']
[Found_Count]
[/inline]
```

SQLite Data Sources

Lasso Professional Server comes with an embedded high-performance data source called SQLite. This data source is used to store Lasso's internal site preferences and security settings.

SQLite is installed, enabled, and pre-configured within Lasso Professional Server by default. No further set up or installation of SQLite is required.

Using SQLite Databases

SQLite databases are stored in the SQLiteDBs folder within each site's folder. By default this folder contains databases that are required for Lasso Professional Server to function, as documented in the *Using Site Administration* chapter.

Custom databases may be created and added to this folder three different ways:

- Creating custom SQLite databases using DatabaseBrowser.lassoapp, which is described in the *Building and Browsing Databases* chapter.
- Issuing SQL statements via the *Utility > SQL* section of Site Administration. For more information on issuing SQL statements, see the *Site Administration Utilities* chapter.
- Creating SQLite databases using a third-party tool and moving them into the SQLiteDBs folder.

Enabling Custom SQLite Databases

Once custom SQLite databases are created or moved to the SQLiteDBs folder, they can be used in Lasso code or tested using the Browse section of the DatabaseBrowser.LassoApp tool.

For information on setting up required Lasso data source permissions for users, see the *Setting Up Security* chapter. For information on setting database, table, and field preferences, see the remaining sections in this chapter.

Other Data Sources

Third-party data sources can be installed in Lasso Professional Server and accessed using the same tools for the built-in data sources and JDBC data sources. Third-party data sources are installed differently depending on their type. Consult the documentation for the data source connector for full details.

- LCAPI data source modules are installed in LassoModules. They usually have an extension of .dylib, .dll, or .so.
- LJAPI data source modules are installed in LassoModules. They usually have an extension of .jar or .class. Any Java libraries required by the module can be installed in JavaLibraries.
- Custom data source modules are installed in LassoStartup. They usually have an extension of .Lasso or .LassoApp.

After a third-party data source is installed Lasso Service must be restarted. The data source will then show up in the **Setup > Data Sources > Connectors** section of Site Administration. Consult the documentation for the data source connector for details about how it should be configured.

Database Preferences

The Databases page lists all databases within a selected data source host. From here, the administrator can enable or disable selected databases, update database settings details, or list tables within a database.

Figure 12: Databases Preferences Page



The Databases Listing panel shows a listing of all the databases available for the current host. The administrator can switch hosts by selecting a new host from the Hosts pull-down menu.

Enabling Databases

Each database must be enabled before it can be used with Lasso. Databases can be enabled in three ways.

- Select a connector and host from the pop-up menus at the top of the *Database Listing* panel. Select the Enable button next to each database which you want to access through Lasso.
- Selecting Enable All will enable all databases shown for the selected host, while selecting Disable All will disable all databases for the selected host. Selecting Enable All New will enable all New databases for the selected host (see the following section for a description of all database status types). Selecting the Refresh button updates the Database Listing panel with the current list of available databases.
- Selecting each database name will show the *Database Detail* panel. See the next section for details about how to modify individual database preferences.

Note: Always select the Refresh button in the Databases Listing panel to view all databases that have been newly opened or loaded in a data source host.

Updating Database Preferences

Selecting a database in the Database Listing panel will show its details in the Database Detail panel, which appears to the right. The following settings can be updated:

- **Alias** – Allows an alternate name to be set for a selected database without changing the actual file name. This is the database name that should be used in Lasso pages, and is the only name that will be associated with that database by Lasso. The alias is the same as the file name by default.
- **Status** – Specifies whether or not a database is accessible by Lasso Professional Server. An Enabled or Disabled status designates that the database has been actively enabled or disabled in Site Administration. Databases are New if the database is open or accessible by the database application at the specified host, but have not yet been enabled or disabled in Site Administration. Databases are Not Available if they exist in Site Administration but are no longer available to the current host. Calling from within a solution any database that has a New, Disabled, or Not Available status will produce an error.
- **Database Username** – (FileMaker Only) The username required for Lasso to connect to the database. This username will only be used if Always Connect with Host Username/Password is set to No.

- **Database Password** – (FileMaker Only) The password required for Lasso to connect to the database. This username will only be used if Always Connect with Host Username/Password is set to No.
- **Always Connect with Host Username/Password** – (FileMaker Only) Specifies whether or not the host-level username and password will be used when connecting to the database. Selecting Yes uses the username and password in the Hosts page. Selecting No uses the username and password set above.
- **SQL Stop Words (SQL Databases Only)** – Allows the administrator to establish a list of words that should not be allowed in SQL statements. For example if Delete is listed as a SQL stop word then no SQL statement containing Delete will be executed on the specified database. Instead, a Lasso Security error will result.

Updating Database Preferences

The administrator can update the database status by entering or selecting an appropriate value for each field, and then selecting the Update button.

Deleting Database Preferences

Selecting the Delete Settings button in the Database Detail panel will remove the database settings from Site Administration, including all permissions, aliases, and options granted to the database, and its corresponding tables and fields.

Database Schema Note: If a database's schema is significantly altered, the recommended method for updating Site Administration is to delete the database settings in the Database Detail panel, and then refresh the Databases Listing panel. Then, all options for the new database schema may be set again in Site Administration.

Current Permissions

The Group Permissions Listing shows which groups have permission to perform actions in the selected table, and which actions are allowed or denied. These can be changed within the Groups section, which is discussed later in this chapter.

Table Batch Change

The table batch change option allows the encoding of every table on the current database to be changed at once. Select a new encoding from the pop-up menu or select Other... and type a custom encoding choice. After clicking perform every table on the current database will be set to the specified encoding.

Note: New tables which are added to the database will not be set to the selected encoding. It may be necessary to adjust the encoding of each new table after they are added or to use the batch change option after new tables are added.

Aliases

Aliases allow databases and tables to be given unique names within Site Administration that are separate from the actual database and table names for each data source. This allows multiple databases with the same names from different hosts to be accessed by Lasso.

The schema of databases can be hidden using aliases, hiding the names of the actual files that contain the database data. On a shared Web server, each database could be given a name specific to the client who has access to the database.

Web site portability can be enhanced using aliases. For example, a site which is being transitional from a FileMaker database Contacts.fp3 to a SQLite database Contacts could use an alias on Contacts so it still appears under the name Contacts.fp3. The alias allows the back-end database to be changed without modifying all of the Lasso pages that reference a given database name.

Once a database or table alias is established, the aliased name must always be used for access to the database through custom Lasso pages. Using a combination of the real name and aliased name is not possible.

If a new host is added to Site Administration and some of the databases have the same names as databases already configured within Site Administration, then the new databases will have aliases created automatically which add an asterisk to their name so it is unique. For example, if a host is added with a Contacts database then it will be added with an alias as Contacts* since its name conflicts with the internal Contacts database.

Note: Aliases are created with an incremental number of asterisks depending on the number of databases with the same name that exist in Site Administration.

Table Preferences

The Tables page lists all tables within a selected database. From here, the administrator can enable or disable selected tables, update table details, or list fields within a table.

Figure 13: Table Preferences Page



The Tables Listing panel shows a listing of all the tables available for the current database. The current connector, host, and database are displayed by default. The administrator can switch databases by selecting a new database from the Database pull-down menu.

Selecting the name of a table shows its details in the Table Detail panel, which appears to the right. Selecting Enable All will enable all tables shown for the selected database, while selecting Disable All will disable all tables shown for the selected database. Selecting Enable All New will enable all tables for the selected database with the status New. Selecting the Refresh button updates the Tables page with the most current information.

Updating Table Status

Selecting a database in the Tables Listing panel will show its details in the Tables Detail panel, which appears to the right. Here, the administrator can update the selected table by modifying its details in the Tables Detail panel, and then selecting the Update button. The following table details can be updated:

- **Alias** – Allows an alternate name to be set for a selected table without changing the actual file name. This is the table name that should be used in Lasso pages, and is the only name that will be associated with that table by Lasso. The alias is the same as the file name by default.
- **Status** – Specifies whether or not a table is enabled for use with Lasso Professional Server. Calling a disabled table from within a solution will produce an error.
- **Encoding** – Specifies the character encoding that will be used for all data stored in each field in the table. This can be ISO8859-1 (default), or UTF-8 (Unicode).

Updating Table Preferences

The administrator can update the table status by entering or selecting an appropriate value for each field and selecting the Update button.

Deleting Table Preferences

Selecting the Delete Settings button will remove the table settings from Site Administration, including all permissions, aliases, and options granted to the table, and its corresponding fields.

Current Permissions

The Group Permissions Listing shows which groups have permission to perform actions in this table, and which actions are allowed or denied. These can be changed within the Groups section, which is discussed later in this chapter.

Field Preferences

The Fields page lists all fields within a selected table. From here, the administrator can enable or disable selected fields, update field details, or list fields within a table.

Figure 14: Field Preferences Page



The Fields Listing panel shows a listing of all the fields available for the current table. The current connector, host, database, and table are displayed by default. One can switch tables by selecting a new table from the Table pull-down menu. Selecting Enable All will enable all fields for the selected table, while selecting Disable All will disable all fields for the selected table.

Updating Field Status

Selecting a field in the Tables Listing panel will show its details in the Field Detail panel, which appears to the right. The administrator can update the field status by entering or selecting an appropriate value for each field and selecting the Update button.

The following field preferences can be set:

- **Status** – Specifies whether or not a field is accessible by Lasso Professional Server. Calling a disabled field from within a solution will produce an error.
- **Allow Searching on this Field** – Specifies whether or not a field can be searched or viewed by users. Fields can be restricted from being searched by setting this preference to No. The default is Yes so that all fields which are Enabled can be used within searches. This preference can be useful to restrict searches to only indexed fields.

- **Allow this Field to be Modified** – Specifies whether or not a field can be updated by users. Fields can be made read-only by setting this preference to No. The default is Yes so that all fields can be modified provided appropriate permissions are granted for the database and table.
- **Require an Exact Match on this Field** – Specifies whether or not an exact value is required when searching, deleting, or updating records. Exact matches are case-sensitive. The equals operator must be used with that field when it is present in a search action. In addition, if any field is defined as exact match, then the -FindAll and -Random actions cannot be used with that database. See the section on *Record-Level Security* below for more information.
- **Filter Search Values for this Field** – Allows the administrator to insert Lasso code that is executed every time a -Search action is performed on the selected field. The code can alter the search request or return a custom error message if the search request does not meet the specified criteria. For more information on filters, see the *Field Filters* section below.
- **Filter Add Values for this Field** – Allows the administrator to insert Lasso code that is executed every time a -Add action is performed on the selected field. For more information on filters, see the *Field Filters* section below.
- **Filter Update Values for this Field** – Allows the administrator to insert Lasso code that is executed every time an -Update action is performed on the selected field. For more information on filters, see the *Field Filters* section below.

Field Filters

Field filters allow Lasso code to be performed when specified Lasso actions are performed. This can be useful for preventing certain types of data from being submitted to specified fields. For example, you can set up profanity filters, or filters to prevent operations which may adversely affect the performance or integrity of your databases (e.g. lengthy search or sort requests).

Field filters can be attached to a specific field for the search, add, or update actions. When the specified action is performed, the code within the field filter will be executed, modifying the value of the data being submitted for the field, or returning a security error.

The code within each field filter can access the value for the field in the variable `Filter_Field`. The value for this variable at the end of the Lasso code will be the value that is submitted to the data source connector for the action. What follows are examples of various useful field filters.

To use an included file as a filter:

Specify an `[Include]` tag within any type of field filter. The code in the included file will be processed for each field that references the file. This allows the same code to be used for many different field filters. The following tag would include a file named `Add_Filter.lasso` in the `Filters` folder at the root of the Web serving folder.

```
[Include: '/Filters/Add_Filter.lasso']
```

To use a custom tag as a filter:

Define a custom tag within a Lasso page in `LassoStartup`. The custom tag can be called within a filter the same as any Lasso tags. The following example calls a custom tag `[MySearchFilter]` with the `Filter_Field` variable as a parameter. It then stores the value of the custom tag back into the `Filter_Field` variable.

```
[Variable: 'Filter_Field' = (MySearchFilter: 'Filter_Field')]
```

To ensure a value is provided when adding a record:

Insert the following code into an add filter in order to ensure that a value is always entered for a specific field. If this code were entered for the field `Full_Name`, then the value `John Doe` would be inserted whenever the field was left blank in the `[Inline]` that performed the add.

```

[If: (Variable: 'Filter_Field') == "]
  [Variable: 'Filter_Field' = 'John Doe']
[/If]

```

To ensure a minimum length for search parameters:

Insert the following code into a search filter in order to ensure that the value being searched for is alphanumeric, and contains at least three characters. If the search parameter does not meet these criteria, then a field restriction error is reported and the search is not performed.

```

[If: (Variable: 'Filter_Field')->Length >= 3]
  [Loop: (Variable: 'Filter_Field')->Length]
    [If: !(String_IsAlphaNumeric: (Variable: 'Filter_Field')->(Get: Loop_Count))]
      [Error_SetErrorCode: (Error_FieldRestriction: -ErrorCode)]
      [Error_SetErrorMessage: (Error_FieldRestriction)]
    [/If]
  [/Loop]
[Else]
  [Error_SetErrorCode: (Error_FieldRestriction: -ErrorCode)]
  [Error_SetErrorMessage: (Error_FieldRestriction)]
[/If]

```

To filter profanity from added or updated records:

Insert the following code into a file named Profanity_Filter.lasso at the root of the Web serving folder. Call this file within the add and update filters for any fields which need to be checked for profanity as follows.

```

[Include: '/Profanity_Filter.lasso']

```

The profanity filter will take a list of banned words and quickly scan through the submitted field values, replacing any of them with the string &#@\$. The list of banned words is defined in the array Banned_Words and is strictly G-rated for demonstration purposes.

```

[Variable: 'Banned_Words' = (Array: 'idiot', 'weirdo', 'moron')]
[Loop: (Variable: 'Banned_Words')->Size]
  [Variable: 'Filter_Field' = (String_Replace: (Variable: 'Filter_Field'),
    -Find=(Variable: 'Banned_Words')->(Get: Loop_Count),
    -Replace='&#@$!')]
[/Loop]

```

This filter can be easily adapted to forbid banned words from being submitted by returning a field restriction error.

To deny search requests with a slow sort request:

In some databases such as FileMaker Pro, a sort request on an unindexed or related field can significantly impact the ability of the application to serve additional database requests. The following filter returns a field restriction error if a search request is entered that requests a sort on the Grand_Total field. This filter should be entered as a search filter.

```

[Sort_Arguments]
  [If: (Sort_FieldItem) == 'Grand_Total']
    [Error_SetErrorCode: (Error_FieldRestriction: -ErrorCode)]
    [Error_SetErrorMessage: (Error_FieldRestriction)]
  [/If]
[/Sort_Arguments]

```

This filter can be easily adapted to check search criteria or other sort criteria, and to deny the search request if any undesirable actions are attempted.

14

Chapter 14

Setting Up Security

This chapter describes the methodology of securing Lasso-based Web sites using Lasso Security and Site Administration, and is divided into the following sections.

- *Introduction* provides an introduction to Lasso Security and introduces concepts described later in the chapter.
- *Setting Up Groups* describes how to set up groups in Lasso, which represent sets of permissions for end users.
- *Setting Up Users* describes how to define custom usernames and passwords in Lasso to be used for end user authentication.
- *Data Host Permissions* describes the data source host permissions which can be assigned to a group, which includes permission to issue SQL queries.
- *Database Permissions* describes the individual database permissions which can be assigned to a group.
- *Table Permissions* describes the individual table permissions which can be assigned to a group.
- *Field Permissions* summarizes the field preferences which can be set for a field and how they are checked.
- *Tag Permissions* describes how substitution, process, and container tags can be secured by assigning permissions to a group.
- *File Permissions* describes the permissions and preferences unique to file manipulation tags.
- *Web Server Security* describes how Lasso Security interacts with the security systems of Web server applications.
- *Record-Level Security* describes how to use field settings to implement record-level security.
- *Security Tips* includes general principles which help ensure a secure Web site.

Introduction

This chapter describes the tools that Lasso Professional Server provides to allow developers to build secure Web sites. This chapter documents the concepts behind Lasso's security, and strategies for setting up security on servers with various purposes. This chapter also documents the *Setup > Security* section of Site Administration, which is the primary interface for configuring Lasso security.

Lasso Security Model

One of the most important features in Lasso Professional Server is the inclusion of a very robust security model that can be used to secure all elements of a Lasso solution. Lasso Security interacts with Web server applications, database applications, Web clients, operating systems, and other CGI products. Lasso provides tools to secure access to each of the different applications that Lasso uses, and also tools to control usage within Lasso.

Lasso Security is comprehensive, and can be used to secure and protect many Web server operations from undesired users. These operations include but are not limited to the following:

- **Database Actions** – Lasso provides tools to restrict the types of actions that can be performed on databases. The same Lasso security model can be applied to all data sources to which Lasso connects.
- **Tag Processing** – Lasso provides tools to restrict the use of Lasso tags both at the server level, and at the user level.
- **File Manipulation** – Lasso provides permissions specific to tags that manipulate the file system local to Lasso Service.
- **File Access** – Lasso can restrict the file suffixes which can be processed by Lasso at the server level.
- **Web Browser Authentication** – Lasso automatically prompts Web browsers for authentication as needed, and provides tools to allow developers to prompt for custom authentication.
- **Web Server Realms** – Lasso provides tags for interacting with Web server realms and administration for file suffix mappings.

Lasso Groups and Users

At the core of Lasso Security are Lasso groups and users. **Groups** represent sets of permissions for various operations, and **Users** represent usernames and passwords (or lack thereof) which can be used to access those permissions.

The power of Lasso Security lies not only in what permissions can be set for a group, but also in how those permissions can be assigned to users. Group permissions can be shared by one or more users, and users may be assigned to one or more groups (how Lasso determines what permissions to use for a user assigned to multiple groups is described later). This multi-dimensional security model allows the most advanced authentication systems to be created for those that desire them.

In Lasso Security, there are three classes of groups and users, which are described below. Understanding these classes and what they can do is the first step in using Lasso Security with maximum power and ease.

Site Administrator

The Lasso Site Administrator has full control over all Lasso-configurable operations by default. Only the Lasso site administrator or server administrator may access Site Administration. The site administrator permissions account is a pre-configured group in Lasso Security that consists of only one username and password set. Site administrator permissions cannot be shared by other users.

Whenever the site administrator is logged in via a Web browser, he or she will instantly have permission to execute all Lasso code and will pass all security checks without being prompted for authentication again.

When using Lasso for the first time, it is important to understand that the end-users who will be accessing Lasso solutions will not have the same permissions as the site administrator by default, therefore some security settings such as database permissions will have to be set before other users may use them.

AnyUser Group and Anonymous Users

Only one configurable group is predefined in Lasso Professional Server, which is the AnyUser group. If an end user of a Lasso Solution is not logged in with a username and password that is defined in Lasso Security, then he or she will inherit the permissions of this group by default. These users are referred to as Anonymous users.

Figure 1: Group Detail



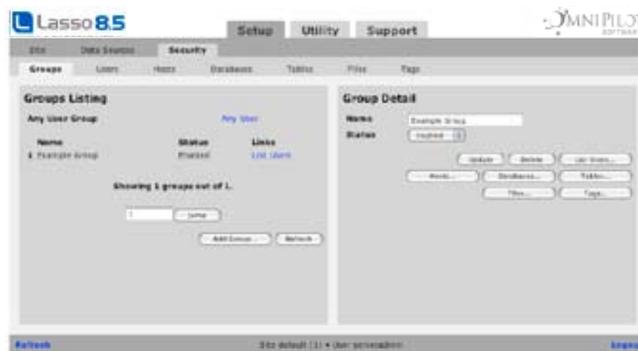
When Lasso Professional Server is installed, it is secure by default. This means that the AnyUser group has no database or file permissions assigned to it in Site Administration by default. When creating a Lasso solution that does not require authentication, explicit permissions must be set for the AnyUser group beforehand. How to set permissions for the AnyUser group is described in this chapter.

Custom Groups and Users

Custom groups may be created by the site administrator if password protection is desired for certain sets of permissions.

A custom group may be created with a defined permissions, and then defined users in Lasso Security (with usernames and passwords) may be added to the custom group. Defined users in Lasso Security may also be added to more than one custom group.

Figure 2: Custom Groups



Much of the security power in Lasso Professional Server comes from creating custom groups and users, and there are many different ways to use groups in Lasso Security to assign permissions to users. Several custom groups with different permissions can be combined to meet the unique needs of any Web site. For example, separate groups could be created for each database used by Lasso, but a single group could define access to file tags across all users of a Web site.

Lasso Professional Server also features the ability to designate group administrators among custom groups. A group administrator is a user in Lasso Security defined by the site administrator who has permissions to make changes to user accounts that belong to a specific group.

User Authentication

All Lasso tags and actions are executed with the permissions of a specific user. Since permissions are only granted to groups, the permissions for a specific user are defined by the group(s) to which they belong. When a tag is executed, the permissions of all the groups to which the user belongs are checked in alphabetical order by group name.

There is no one-to-one correspondence between users and visitors. It is possible for many visitors to authenticate as the same user. For example, all the members of a workgroup may share a common username and password. In the extreme, all unauthenticated visitors are treated as a single Anonymous user and have the permissions of the AnyUser group.

A single visitor may execute code using the permissions of many users. They will be prompted for a username and password if they do not have permission to access a resource based on the current user information they have supplied. If they enter the authentication information for a different user, then they become that user for subsequent pages.

Lasso primarily relies on the Web authentication routines built into every Web browser in order to authenticate site administrators and other Lasso users. However, Lasso permissions may also be programmatically specified using the `[Inline] ... [/Inline]` tag in Lasso.

Username and passwords can be specified within `[Inline] ... [/Inline]` tags so that the contents of the container tags are executed with the permissions of a specified user. Most of a Lasso page will be executed with the permissions of the user defined by the current visitor, but portions of the Lasso page can be executed with the permissions of specific users.

There are several ways to specify what user's permissions should be used to execute a Lasso page or section of Lasso code depending on whether or not the current visitor is authenticated, or if a specific username and password are coded into the Lasso page.

- **Anonymous** – By default, all code is executed as the Anonymous user. This is the case if no users are defined within Lasso Security, if the current visitor has not specified a username and password, or if the current visitor has specified an invalid username and password.
If an Anonymous user does not have permission to access a resource, then an authentication dialog will be displayed in the visitor's Web browser. If one enters a valid username and password for a user that has permission to access the resource, then one can continue.
- **Current Visitor** – If the current visitor has entered a username and password into the Web browser's authentication dialog and that username and password can be found in Lasso Security, then the Web site will execute all tags with the permissions of that user.
- **Inline Authentication** – The `-Username` and `-Password` command tags can be used to explicitly set a username and password for all code executed within `[Inline] ... [/Inline]` tags (including a database action within the opening `[Inline]` tag). This allows the permissions of the current visitor to be augmented by the permissions for a specific user for the duration of the `[Inline] ... [/Inline]` tags.
- **Site Administrator** – A single site administrator is defined by Lasso Security. This user has the authority to execute any tag and perform any database action even on tags and databases which are disabled. This user also has the authority to change all of the security settings within Site Administration. The site administrator username and password should never be embedded in a Lasso page.

How Lasso Checks User Security

All of Lasso Security is tied to the action of executing Lasso tags. Lasso performs a series of checks each time a tag is executed in order to confirm that the current user has permission to execute the tag, and the current user has permission to access the resources that the tag requires in order to perform its task.

The process that Lasso uses to check security settings each time a tag is executed is shown below. Some of the steps are actually performed when a Lasso page is first accessed, or when an `[Inline] ... [/Inline]` is entered and information is cached for fast access, but conceptually the steps are performed in this order.

When a user belongs to more than one group in Lasso Security, groups are always checked in alphabetical order with the exception of the AnyUser group, which is always checked last. Groups with names that start with letters earlier in the alphabet have higher priority than groups with names that start with letters later in the alphabet. If a specific ordering of groups is needed, then a prefix can be used to ensure that the group permissions are checked in the desired order.

How Lasso checks security when a Lasso action is performed:

- 1 The list of available Lasso tags is checked. Any tags which are not enabled (e.g. tags marked Disabled) are not recognized as part of Lasso, and will generate a syntax error (rather than a security violation).
- 2 The current username and password are identified. If the current site visitor has authenticated through their Web client, then the username and password they specified are used. If -Username and -Password are specified within the Lasso code, then the values of those tags are used. Otherwise, the user is Anonymous.
- 3 The current user is looked up in Lasso Security. If no matching username and password can be found, then the user is treated as Anonymous. Note that Lasso will execute code for a visitor who has specified any unknown username and password as Anonymous.
- 4 The set of groups which the current user belongs to is identified. The permissions for each group will be checked in alphabetical order, followed by the AnyUser group.
- 5 The permissions for each group are checked in alphabetical order. If an Allow permission is found for the tag, then its use is allowed and no further checking is performed. If a Deny permission is found for the tag, then its use is denied, a security violation is generated, and no further checking is performed. If an Ignore permission is found, then the use of the tag is not allowed for that group, but Lasso will continue to check that tag's permissions for the current user in other groups. By default, all substitution tags in AnyUser have Allow permission, while all action tags (e.g. -Search, -Add, -Update, -Delete, -SQL) have Ignore permission.
- 6 If the tag references a database, table, or fields, then the permissions specific to those entities are checked. See the sections on *Database Permissions*, *Table Permissions*, and *Field Permissions* for more details.
- 7 If the tag is a [File_...] tag, then the permissions specific to the [File_...] tags are checked. Site users and the site administrator can only use the file tags within the paths defined in Server Administration. See the *File Permissions* section for more information.
- 8 If a security violation is generated, then the site visitor will be presented with a browser-specific authentication dialog. After the visitor has entered a new username and password, the entire series of steps are performed again checking against a new user.

Security Error Reporting

Unless custom error handling is coded into your .lasso or .lassoapp files, the Lasso Professional default error page will be shown in the event of a security error while processing a file. The level of detail shown in the default page can be configured using the **Setup > Site > Syntax** section of Site Administration.

Important: A configured error reporting of Full in the **Setup > Site > Syntax** section of Site Administration will reveal Lasso source code and other critical information when an error occurs. While this is useful for debugging, it is strongly recommended that Lasso Professional servers running in a production environment be set with a error reporting level of Minimal or None for maximum security.

To learn more about error reporting in Lasso Professional and how to code custom error handling using Lasso, see the *Error Control* chapter in the Lasso 8.5 Language Guide.

Setting Up Groups

Groups in Lasso Professional Server allow security settings to be set for groups of users. Users can log in to a Lasso-driven Web site with assigned group security values, or in the absence of a recognized username and password are automatically assigned to the AnyUser group by default.

The *Setup > Security > Groups* page in Site Administration allows the administrator to add, view, enable, disable, and configure groups in Site Administration.

Figure 3: Groups List Page



The Group Listing panel shows a list of all the groups defined in Lasso. Prev and Next buttons are shown if more than 10 groups exist. Selecting Enable All will enable all groups shown, while selecting Disable All will disable all groups shown. Selecting Refresh will refresh the Groups page with the most recently updated information.

The default group is AnyUser with one member Anonymous. By default, any permissions granted to this group will be granted to all visitors to the Web site.

Adding a Custom Group

Selecting the Add Group button shows the Add Group panel to the right. The Add Group panel allows the administrator to add a new group to Site Administration.

Figure 4: Add Group Panel



To add a new group:

- 1 Enter the name of the group to be added in the Name field.
- 2 Select Enabled or Disabled from the Status pull-down menu to enable or disable the group in Lasso Professional Server.
- 3 Select Add Group.

After selecting the Add Group button, one is taken to the Groups page with the newly added group selected for additional modifications.

Updating Group Properties

Selecting a group name in the Groups Listing panel will show its details in the Group Detail panel, which appears to the right. Here, the administrator can update the selected group name and status by selecting the Update button, or delete the group by selecting the Delete button. The AnyUser group is separated from all created groups, and appears at the top of the Groups Listing panel.

To update details for a selected group:

- 1 Edit the name of the selected group in the Name field.
- 2 Select Enabled or Disabled from the Status pull-down menu to enable or disable the group.
- 3 Select Update.

The List Users link takes one to the Users page with the current group selected. The date and time of the last update is always shown in the lower right corner of the Group Detail panel.

AnyUser Note: The AnyUser group cannot be deleted as it is required by Lasso Professional Server. For this reason, the Update and Delete buttons will not be shown in the Group Detail panel for the AnyUser group.

Setting Group Permissions

Group permissions can be set by selecting the Hosts, Databases, Tables, Tags, or Files buttons in the Group Detail panel. Doing so will take one to the Hosts, Databases, Tables, Tags, or Files pages where the corresponding permissions can be set, as discussed later in this chapter.

For most Lasso Solutions, setting database and table permissions for the AnyUser group is typically the first task. See the *Data Host Permissions*, *Database Permissions*, and *Table Permissions* sections for more information on these permissions.

Setting Up Users

Individual user accounts allow different users to access Lasso Professional Server Web solutions with different levels of security. While access and tag-level security settings are set in the Groups section, the Users section allows the administrator to set usernames and passwords, set groups to which a user belongs, and to designate specific users as administrators for specific groups.

The *Setup > Security > Users* page in Site Administration allows the administrator to add, view, and configure users.

Important: No user configuration is required for Anonymous users, which are configured by default through the AnyUser group. The Users page is used only for creating custom username and password accounts, and designating custom group administrators.

Figure 5: Users List Page



User Listing

The User Listing panel lists all currently defined users. Users from a particular group as well as users who do not belong to any groups can be shown. The current group may be changed by selecting a group from the pull-down menu. In addition to the groups defined in the *Setup > Security > Groups* section of Site Administration, the administrator may choose one of the following user display options from the pull-down menu.

- **All Users** – Displays all users in Site Administration regardless of group assignment.
- **No Group** – Displays all users not currently assigned to groups.
- **All Admins** – Displays all group administrators in all groups.

Selecting the username shows details of the user in the User Detail panel, which appears to the right. All users shown can be enabled or disabled by selecting Enable All or Disable All. Selecting Add User... takes one to the Add User page.

Adding Custom Users

Selecting the Add User button in the User Listing panel shows the User Detail panel to the right. The Add User page allows the administrator to create new users in a specified group or groups.

Figure 6: Add User Panel



To add a new user:

- 1 Enter the username of the user to be added in the Name field.
- 2 Enter the password for the selected user in the Password field.
- 3 Select any number of groups from the Groups field to which the user is to be assigned.
- 4 Select any number of groups from the Administrator field for which the selected user is to be granted administrator privileges. The purpose of a group administrator is to add and remove other users to and from the group the user has group administrator privileges for. There can be as many group administrators as needed. For more information on group administrators, see the following *Group Administrators* section.
- 5 Select the Add User button.

The Search page allows administrators to search for users by group, name, status, or administrator.

Updating Custom Users

Selecting a user in the User Listing panel will show its details in the User Detail panel, which appears to the right. Here, the administrator can update or delete the selected user's details by selecting the Update button, or delete the user by selecting the Delete button.

To update details for a selected user:

- 1 Edit the username of the selected user in the Name field.

- 2 Edit the password for the selected user in the Password field. The password will be hidden, but can be changed by deleting the contents of the Password field, and entering a new password.
- 3 Select Enabled or Disabled from the Status pull-down menu to enable or disable the user.

Note: Disabled users are treated as anonymous users, and are assigned permissions from the AnyUser group when accessing Lasso pages or LassoApps.
- 4 Select any number of groups from the Groups field to which the user is to be assigned.
- 5 Select any number of groups from the Administrator field for which the selected user is to be granted administrator privileges. The purpose of a group administrator is to add and remove other users to and from the group the user has group administrator privileges for. There can be as many group administrators per group as needed. For more information on group administrators, see the following *Group Administration LassoApp* section.
- 6 Select the Update button.

Note: The Anonymous user cannot be modified or deleted as it is required by Lasso Professional Server. For this reason, the Delete button will not be displayed in the User Detail field for the Anonymous user.

Group Administrators

Users can be granted the privilege to be a group administrator for one or more groups. Any user which is defined as a group administrator can add users to a group using the GroupAdmin.LassoApp file, or using the [Admin_...] tags in Lasso. Only a group administrator or the site administrator can add users to a group, and a group can have as many group administrators as needed.

Group Administration LassoApp

The group administrator is able to add and remove users from a group via the GroupAdmin.LassoApp file, which is a LassoApp served on-demand from the Lasso Professional Server application folder. This LassoApp only contains the features of the *Setup > Security > Users* section of Site Administration. When accessed, the GroupAdmin.LassoApp file prompts one for a group administrator username and password, as designated in the *Setup > Users > Admin* page in Site Administration. Within the Group Administration LassoApp, group administrators are only able to access the group to which they have been assigned.

Figure 7: Group Administration LassoApp



To access the Group Administration LassoApp:

In a Web browser, visit <http://www.example.com/Lasso/GroupAdmin.LassoApp>. Replace www.example.com with your domain name, IP address, or 127.0.0.1 if on a local machine. If an error is displayed, make sure Lasso Service is running.

Data Host Permissions

The *Setup > Security > Hosts* page in Site Administration allows the administrator to define all host-level permissions for any specific group.

Figure 8: Hosts Page



The top panel shows the current group to the right of Permissions For Group. The current group can be switched by selecting an existing group from the Groups pull-down menu and selecting Switch Group.

The Host Listing panel shows a listing of all the data source hosts available for the current group. The current connector is displayed by default. The administrator can switch connectors by selecting a new connector from the Connector pull-down menu. Selecting Unassign All will reset all permissions for all hosts to their default values.

Setting Group Host Permissions

Selecting a host in the Host Listing panel shows its permissions settings in the Host Detail panel to the right. The following describes the database security permissions that can be set for the selected group:

- **Execute SQL** – Allows or denies the ability to issue SQL statements by restricting the -SQL command tag in Lasso 8. Allowing Execute SQL effectively grants a group the ability to view and edit all databases, tables, and fields in a host via SQL statements, and should be allowed with care. This is the only permission specific to the host-level in Lasso Professional Server, and is only applicable for data sources that can interpret SQL statements.

Warning: Due to the large array of things that permission to execute SQL statements opens up for a developer, SQL permission should only be given to trusted users.

The administrator can update host security settings by selecting Allow or Deny from the Execute SQL pull-down menu, and then selecting the Update button. The List Databases button takes the administrator to the Databases page with the current host selected.

How Lasso Checks Data Host Security Settings

When a user attempts to perform a SQL query via Lasso, the permissions for all the groups the user belongs to are checked in sequence. The following steps are performed.

- 1 The set of groups which the current user belongs to is identified. If the user belongs to more than one group, the permissions for each group will be checked in alphabetical order by group name.
- 2 The permissions for each group are checked for a permission which references the current host. If an Allow permission is found for the host, then its use is allowed and no further checking is performed. If a Deny permission is found for the host, then its use is denied, a security violation is generated, and no further checking is performed. If an Ignore permission is found, then the database is not allowed for that group, but Lasso will continue to check database permissions for the current user in other groups.
- 3 If no result is found after checking all of the groups which the user belongs to and the AnyUser group, then the SQL query is denied and a security violation is generated.

- If the database referenced in the SQL query has any SQL stop words defined in Site Administration, they are compared to the SQL statement which was specified in the Lasso SQL query. If a match is found, then the SQL statement is not allowed and a security violation is generated.

An Allow permission is required in order to issue SQL statements for a data source host. The SQL permission for a host is Ignore by default, and this permission must be explicitly set to Allow before any SQL statements can be issued using Lasso.

Database Permissions

Lasso Security allows administrators to either Allow or Deny different database actions for a group for each specific database available through Lasso. Permissions can also be defined for a single meta-entry **All Databases**. This entry is checked if no permissions are found for a specific database.

The *Setup > Security > Databases* page in Site Administration allows the administrator to define all database-level permissions for any specific group.

Figure 9: Databases Page



The top panel shows the current group to the right of Permissions For Group. The current group can be switched by selecting an existing group from the Groups pull-down menu and selecting Switch Group.

The Database Listing panel shows a listing of all the databases available for the current group, as well as links to All Databases and All Tables settings. The current connector and host are displayed by default. The administrator can switch connectors and hosts by selecting a new connector and host from the Connectors and Hosts pull-down menus. Selecting Unassign All will reset all permissions for all databases to their default values. No user will be able to access a database via a Lasso solution unless the database has been assigned with at least one permission set to Allow by the administrator.

Selecting the All Databases link in the Database Listing panel allows permissions to be set for all databases regardless of data source connector or host in the Database Detail panel, which appears to the right. Selecting the All Tables link takes the administrator to the Table Detail panel in the Tables page, where permissions for all tables regardless of data source connector, host, or database can be set. For more information, see the Tables page description.

Setting Group Database Permissions

Selecting a database in the Databases Listing panel will show its permissions settings in the Database Detail panel for the selected group. Here, the administrator can update database security settings by selecting the Update button.

The following describes the database security permissions that can be set for the selected group:

- **Search Records** – Allows or denies record searching for the selected group by restricting the -Search, -FindAll, and -Random command tags in Lasso 8.
- **Add Records** – Allows or denies the ability to add records for the selected group by restricting the -Add tag.
- **Update Records** – Allows or denies the ability to update records for the selected group by restricting the -Update tag.
- **Delete Records** – Allows or denies the ability to delete records for the selected group. This is accomplished by restricting the -Delete command tag in Lasso 8.
- **Show Database Schema** – Allows or denies the ability to view a database’s schema for the selected group. The database schema is a list of all tables in a database, and is accomplished by restricting the -Show command tag in Lasso 8.
- **Execute Scripts** – Allows or denies the ability to execute FileMaker Pro scripts utilizing the -FMScript command tags in Lasso 8.

Note: All permissions default to **Ignore**, which simply is the absence of an **Allow** or **Deny** permission.

The administrator can update database security settings by selecting **Allow** or **Deny** from each pull-down menu, and then selecting the **Update** button. The **List Tables** button takes the administrator to the **Tables** page with the current table selected.

How Lasso Checks Database Security Settings

When a user attempts to perform a database action, the permissions for all the groups the user belongs to are checked in sequence. The following steps are performed.

- 1 The set of groups which the current user belongs to is identified. If the user belongs to more than one group, the permissions for each group will be checked in alphabetical order by group name.
- 2 The permissions for each group are checked for a permission which references the current database and action. If an **Allow** permission is found for the database, then its use is allowed and no further checking is performed. If a **Deny** permission is found for the database, then its use is denied, a security violation is generated, and no further checking is performed. If an **Ignore** permission is found, then the database is not allowed for that group, but Lasso will continue to check database permissions for the current user in other groups.
- 3 If no result is found in a particular group, then the permission for the **All Databases** entry for that group is checked. If an **Allow** permission is found for the database action for **All Databases**, then the action is allowed. If a **Deny** permission is found, then the database action is denied and a security violation is generated. Note that after the first **Allow** or **Deny** is found, no further checking is performed.
- 4 If no result is found after checking all of the groups which the user belongs to and the **AnyUser** group, then the database action is denied and a security violation is generated.
- 5 If the database has any SQL stop words defined, they are compared to the SQL statement which was specified in the -SQL command tag or to the SQL statement generated automatically by Lasso. If a match is found, then the SQL statement is not allowed and a security violation is generated.

This order of operation means that any database actions which are not explicitly allowed for a group (and its associated users) are effectively denied. The **All Databases** entry can be used to allow database actions across all databases available to Lasso.

Database permissions are required in order to access a table. A database must have **Allow** permission for a specified action in order for the like permissions of the table to even be checked. All actions for a table have **Ignore** permission by default, and permissions must be explicitly set to **Allow** before any data can be retrieved from the table.

Table Permissions

Lasso Security allows administrators to either Allow or Deny different database actions for a group for each specific table available through Lasso.

The *Setup > Security > Tables* page in Site Administration allows the administrator to define all table-level permissions for any existing group.

Figure 10: Tables Page



The top panel shows the current group to the right of Permissions For Group. The current group can be switched by selecting an existing group from the Groups pull-down menu and selecting Switch Group.

The Table Listing panel shows a listing of all the tables available for the selected database. The current connector, host, and database are displayed by default. The administrator can switch databases by selecting a new database from the Database pull-down menu. Selecting the name of a table shows its group security settings in the Table Detail panel. Selecting Unassign All will reset all permissions for all tables to their default values.

Selecting the All Databases link in the Table Listing panel takes the administrator to the Databases page, where permissions for all databases can be set in the Database Listing panel. Selecting the All Tables link allows permissions to be set for all databases regardless of data source connector, host, or database in the Table Detail panel, which appears to the right.

Note: Allowing permission for all tables does not override any database permission setting of Deny or Ignore. If a database permission is set to Deny or Ignore, then a user will not be able to access any information in the database even if all table permissions are set to Allow.

Setting Group Table Permissions

The administrator can update table security settings for the selected group in the Tables Detail panel. The following describes the table security permissions that can be set for the selected group.

- **Show Table Schema** – Allows or denies the ability to view a table’s schema for the selected group.
- **Search Records** – Allows or denies record searching for the selected group.
- **Update Records** – Allows or denies the ability to update records for the selected group.
- **Add Records** – Allows or denies the ability to add records for the selected group.
- **Delete Records** – Allows or denies the ability to delete records for the selected group.

Note: In order for a user to be able to use a database via a Lasso solution, permissions must be set to Allow on both the database and table level. For example, a user will not be able to search for records in a database unless the Search Records permission is set to Allow for both the database and the table that contains the records.

The administrator can update database security settings by selecting Allow or Deny from each pull-down menu, and then selecting the Update button.

How Lasso Checks Table Security Settings

When a user attempts to perform a database action, the permissions for all the groups the user belongs to are checked in sequence. The following steps are performed.

- 1 The set of groups which the current user belongs to is identified. If the user belongs to more than one group, the permissions for each group will be checked in alphabetical order by group name.
- 2 First, permission is checked for the database which contains the table and for the *All Databases* entry. If *Allow* permission is granted, then the table security settings are checked.
- 3 The permissions for each group are checked for a permission which references the current table and action. If an *Allow* permission is found for the table, then its use is allowed and no further checking is performed. If a *Deny* permission is found for the table, then its use is denied, a security violation is generated, and no further checking is performed. If an *Ignore* permission is found, then the use of the table is not allowed for that group, but Lasso will continue to check table permissions for the current user in other groups.
- 4 If no result is found in a particular group, then the permission for the *All Tables* entry for that group is checked. If an *Allow* permission is found for the database action for *All Tables*, then the action is allowed. If a *Deny* permission is found, then the database action is denied and a security violation is generated. Note that after the first *Allow* or *Deny* is found, no further checking is performed.
- 5 If no result is found after checking all of the groups which the user belongs to and the *AnyUser* group, then the database action is denied and a security violation is generated.

Database permissions are required in order to access a table. A database must have *Allow* permission for a specified action in order for the permissions of the table to even be checked.

Note: In order for a user to be able to use a database via a Lasso solution, permissions must be set to *Allow* on both the database and table level. For example, a user will not be able to search for records in a database unless the *Search Records* permission is set to *Allow* for both the database and the table that contains the records.

Field Permissions

Field permissions are established as preferences in the *Setup > Data Sources > Fields* page of Site Administration. Field permissions are discussed in the *Setting Up Data Sources* chapter, and also in the *Record-Level Security* section that follows.

If both the database and table referenced by a given Lasso action have been allowed, then the field preferences are checked. Since field preferences cannot be set on a per-group basis, they are the same for all users who have access to the database and table regardless of what groups they belong to.

How Lasso checks field permissions:

- 1 The *Allow...* preferences are checked to ensure the field can be used in searches, or can be modified. Otherwise, a security violation is generated.
- 2 The *Require an Exact Match...* preferences are checked and if any are set to *Yes*, the values of the fields involved are compared to those in the database before the action is allowed. If any field values do not match, then a security violation is generated.
- 3 The field filters are processed. If any of them return errors, then the error is returned and the database action is not performed. Otherwise, the values of the field filters are used to process the database action.
- 4 If the field is being referenced as a response field, then the appropriate preferences must be established for the field. Otherwise, a security violation is generated.

Tag Permissions

Lasso Security allows administrators to either Allow or Deny access to any of the tags in Lasso. The *Setup > Security > Tags* page in Site Administration allows the administrator to define all tag-level permissions for any existing group.

Figure 11: Tags Page



The top panel shows the current group to the right of Permissions For Group. The current group can be switched by selecting an existing group from the Groups pull-down menu, and then selecting Switch Group.

Setting Tag Category Permissions

The Tag Categories Listing panel shows a listing of all relevant tag categories for which tags can be enabled or disabled at the group level. Category level permissions can be set by selecting Allow or Deny from the Status pull-down menu, and selecting Update. All categories can be allowed or denied by selecting Allow All or Deny All. A Mixed status indicates that a selected tag category may contain both Allow and Deny permissions for individual tags.

Setting Tag Permissions

When a category is selected in the Tag Categories Listing panel, it shows all relevant tags in that category and their security permissions for the selected group in the Tags Listing panel. Prev and Next buttons appear for navigation if there are more than ten tags in a selected category. Each tag can be individually allowed or denied by selecting Allow or Deny from the Status pull-down menu, and then selecting Update. An Ignore permission designates the absence of an Allow or Deny permission. All tags shown in the Tags Listing panel can also be allowed or denied by selecting Allow All or Deny All.

Tag Dependencies

Some Lasso tags have dependencies on other tags to function properly. If permissions for a tag has been denied that has dependencies with other tags, then those tags will not function correctly. For example, if [TCP_...] tags are denied, it will cause tags such as [Email_Send] not to work, which relies on the use of the [TCP_...] tags. To see whether on not a tag has any dependencies, consult the Lasso 8 Reference and check to see if there are any tags listed in the Required field in the Detail view for any tag.

When a user attempts to execute a tag, the permissions for all the groups they belong to are checked in sequence. The following steps are performed.

How Lasso Checks Tag Security Settings

- 1 The set of groups which the current user belongs to is identified. If the user belongs to more than one group, the permissions for each group will be checked in alphabetical order by group name.
- 2 The permissions for each group are checked for a permission which references the current tag. If an Allow permission is found for the tag, then the tag execution is allowed. If a Deny permission is found, then the tag execution is denied and a security violation is generated. Note that after the first Allow or Deny is found, no further checking is performed.
- 3 If no result is found, then the tag execution is allowed.

This order of operation means that any tag which is not explicitly denied is implicitly allowed.

File Permissions

Lasso Security allows administrators to establish permissions for what files a group of users can manipulate, and what operations they can perform on those files.

The *Setup > Security > Files* page in Site Administration allows the administrator to set file-level security for selected groups. This affects the use of the [File_...], [HTTP_...], [FTP_...], [Image], and [PDF_...] tags, which are described in the Lasso 8.5 Language Guide and Lasso 8 Reference.

Figure 12: Files Page



The top panel shows the current group to the right of Permissions For Group. The current group can be switched by selecting an existing group from the Group pull-down menu, and then selecting Switch Group.

Viewing File Permissions

The File Permissions panel shows what file permissions the current group has. Each entry has two parts.

- **Path** – The path in which a user can access files. Any files in sub-folders of this path can be accessed. A path starting with / is located within the Web server root. A path starting with /// or C:// is located outside the Web server root.

The paths listed are all established by the server administrator for this site. By default permission is granted to access the Web server root and the file uploads location.

- **Permissions** – Lists the permissions that a user in the current group has for files within the path. The meaning of the permissions are listed in full on the following page.

Selecting any path shows information about the path in the *Path Permissions* panel to the right. The permissions can then be changed according to the instructions in the following section.

Note: The final path / (Read Only) is included to indicate that permission is always granted for users to read any files within the Web server root.

The *Unassign All* button at the bottom of the panel unassigns all permissions for all paths for the current group. The *Refresh* button simply refreshes the current panel.

Updating File Permissions

The File Permissions panel shows the file permissions for a single path. The path can be changed or the file permissions for that path modified.

- **File Root** – Designates the folder that the group may access. Since the available roots are determined by the server administrator the root cannot be edited here.
- **Inspect Files** – Allows users in the group to retrieve information about files, such as the file size or creation date.
- **Read Files** – Allows users in the group to read the content of existing files.
- **Create Files** – Allows users in the group to create new files.
- **Write Files** – Allows users in the group to write to existing files.
- **Move Files** – Allows users in the group to move existing files.
- **Copy Files** – Allows users in the group to copy existing files.
- **Delete Files** – Allows users in the group to delete existing files.
- **Any File Extensions** – Allows users in the group to access files with any file permission, not just those which are explicitly allowed.

File permissions can be set by selecting *Allow* or *Deny* for each permission, and selecting *Update*. All permissions can be allowed or denied by selecting *Allow All* or *Deny All*.

File Uploading

In order for site developers to be able to use uploaded files, they must be granted permission to access the temporary folder in which uploaded files are placed. They usually must also be granted permission to access a folder where the uploaded files will be copied permanently.

The location of the temporary items folder can be different depending on how the operating system is set up. The server administrator needs to add the file uploads path to the list of available paths in order to allow group permission to be assigned to it.

Lasso automatically deletes any files which are not removed from the temporary items folder before the current Lasso page is finished being processed. This ensures that unwanted uploads are not allowed to occupy disk space for longer than the time it takes Lasso to process a single Lasso page.

In addition, the temporary items folder is automatically cleaned by the operating system when the computer is restarted. This ensures that even if Lasso Service crashes, unwanted file uploads will be deleted the next time the computer is restarted.

See the *Files and Logging* chapter in the Lasso 8.5 Language Guide for more information about using file uploads.

How File Permissions are Checked

- 1 When a file tag is called, the groups for the current user are determined. In the following example, `[File_Read]` attempts to read a file within the Web server root in a Windows machine.

```
[File_Read: 'c://inetpub/wwwroot/default.lasso']
```

- 2 The permissions for the groups are checked in alphabetical order by group name for a File Root that contains the file which is referenced within the `[File_Read]` tag. For example, a group that provides access to the `wwwroot` directory would match the `[File_Read]` tag above.

```
Allow Path = 'c://inetpub/wwwroot/'
```

- 3 Once a matching File Root is found, the individual permissions are checked. For a [File_Read] tag, the user must be granted Read Files permission.
- 4 If Read Files is not permitted, then the permissions for the groups continue being checked for another permission that matches the File Root of the parameter to [File_Read]. If no valid permission can be found then a security violation is generated.

Web Server Security

Most Web server applications provide built-in security which protects files with certain names or in certain directories. Lasso generally works in concert with Web server security, but can also be used in some instances to bypass Web server security.

- **Lasso Pages** – If Lasso pages are called directly, they will only be served if the current site visitor has permission to access the specified file. For example, if the Web server has a realm protecting the folder `Private` then the following URL can only be accessed if the visitor has permission to access that realm.

`http://www.example.com/Private/default.lasso`

Lasso will be called on to process the Lasso page only if the appropriate file suffix (e.g. `.lasso`) is set properly within the Web server administration options. The file suffix must also be allowed within Lasso Security.

- **Response Files** – Lasso pages which are referenced using one of the `-Response...` tags may obey Web server realms. Depending on the Web server application, Lasso checks with the Web server's security before serving any file as a response page. Since Lasso can work with many different Web Server connectors it is best to test whether realms protect Lasso response files before relying on this security.

For example, if the Web server has a realm protecting the folder `Private` then the following URL can only be accessed if the visitor has permission to access that realm.

`http://www.example.com/Action.Lasso?-Response=/Private/default.unknown`

The suffix mappings within the Web server administration options do not have any effect on what files Lasso will process using a `-Response...` tag. The `.lasso` suffix in `Action.Lasso` ensures that Lasso is called without checking the file suffix of the actual response file.

This allows Lasso pages with suffixes not defined in the Web server administration options to be processed by Lasso. However, the file suffix settings in Lasso Security do still apply. Lasso will only process Lasso pages with suffixes that have been allowed in Lasso Security. In this case, the suffix `.unknown` would have to be allowed in Lasso Security.

- **Include Files** – Lasso pages which are served as part of a valid Lasso pages using one of the `[Include...]` tags are not dependent on the Web server's file suffix mappings or realms. Any file within the Web server root which is allowed by Lasso's own file suffix settings within Lasso Security can be served using the `[Include...]` tag.

Important: The `[Include...]` tags can be used to bypass Web server realms and serve any files from within the Web server root. Only trusted developers should be allowed to upload Lasso pages.

- **File Tags** – The file tags can be used to manipulate any files within the `Allow Path` according to the permissions which have been granted to the current group. It is important to grant only the least number of permissions which each developer needs to implement the desired behavior of their site.

If `Allow Access to Files Outside Root` is granted to a group, then any users in that group can access any files on the machine hosting Lasso Service. This includes any files on the machine inside or outside of the Web server root.

Record-Level Security

Record-level security can be enabled in Lasso by setting fields to require an exact match when various Lasso actions are performed. A specific action will be allowed only if a value is specified for each field that requires an exact match, and if that value is exactly the same as the value currently stored in the database.

Note that record-level security will only be checked if the current user has permission to perform the action in the database or table directly. Record-level security can only be used to deny actions to users who would otherwise be able to perform an action.

Three types of actions can be secured using record-level security.

- **Search** – Records can only be viewed when the proper value for a field is specified. This could be used to allow visitors to password protect individual records. No other visitors will be able to view a specific record unless they know the proper password.
- **Update** – Records can be updated only when the proper value for a field is specified. This could be used to allow only authorized visitors to update certain records. No other visitors will be able to update a specific record unless they know the proper password. This can be used separate from the search option in order to allow visitors to view records, but not to update them.
- **Delete** – Similarly, records can be deleted only when the proper value for a field is specified. This can secure a database so only certain users can delete records, or can serve as a simple check where deletes must be confirmed by specifying a special field value before they are carried out.

To establish record-level security:

- 1 Create two fields within the database or table to be secured. In the following example, Username and Password fields will be created in the People table of the Contacts database. Visitors will be able to specify their username and password when they create a new record. Only the visitor that created the record will then be able to update, or delete that record, but anyone will be able to view the record.
- 2 Within Site Administration *Setup > Data Sources > Fields* check the box for Require an Exact Match on this Field when Updating Records for both the Username and Password fields.
- 3 Ensure that the AnyUser group has permission to search the database, add records to the database, and to update records within the database. The exact match option will only be checked when the user attempts to update a record within the database.
- 4 The add records page needs to require values for both the Username and Password fields. The following example shows an HTML form with `-Required` command tags embedded before the text inputs for Username and Password.

```
<form action="response.lasso" method="post">
  <input type="hidden" name="-Required" value="">
  <p>Username: <input type="text" name="Username" value="">
  <input type="hidden" name="-Required" value="">
  <p>Password: <input type="text" name="Password" value="">
  <p>First Name: <input type="text" name="First_Name" value="">
  <p>Last Name: <input type="text" name="Last_Name" value="">
  <p>Phone: <input type="text" name="Phone_Number" value="">
  <p><input type="submit" value="Add Record">
</form>
```

When this form is submitted, the following `[Inline] ... [/Inline]` tags in the `response.Lasso` page add a new record to the table with the Username and Password established for that record.

```
[Inline: -Add,
  -Database='Contacts',
  -Table='People',
  -KeyField='ID',
  'Username'=(Action_Param: 'Username'),
  'Password'=(Action_Param: 'Password'),
  'First_Name'=(Action_Param: 'First_Name'),
  'Last_Name'=(Action_Param: 'Last_Name'),
  'Phone_Number'=(Action_Param: 'Phone_Number')]
[/Inline]
```

- 5 Any user will be able to view the First_Name, Last_Name, and Phone_Number established for the user using simple search and response Lasso pages. Of course, the Username and Password fields should never be revealed on any search response.
- 6 Users will be able to modify their record in the database using the following update form. This form is similar to the add form which was defined above, but has the added feature of the exact match setting. Lasso Security will not allow the [Inline] ... [/Inline] which this form triggers to be processed unless both the Username and Password fields are defined with the right values.

```
<form action="response.lasso" method="post">

  <input type="hidden" name="-Required" value="">
  <p>Username: <input type="text" name="Username" value="">

  <input type="hidden" name="-Required" value="">
  <p>Password: <input type="text" name="Password" value="">

  <p>First Name: <input type="text" name="First_Name"
    value="[Field: 'First_Name']">
  <p>Last Name: <input type="text" name="Last_Name"
    value="[Field: 'Last_Name']">
  <p>Phone: <input type="text" name="Phone_Number"
    value="[Field: 'Phone_Number']">

  <p><input type="submit" name="-Nothing" value="Update Record">
</form>
```

When this form is submitted, the following [Inline] ... [/Inline] tags in response.lasso perform the update action within the database.

```
[Inline: -Update,
  -Database='Contacts',
  -Table='People',
  -KeyField='ID',
  -KeyValue=(KeyField_Value)
  'Username'=(Action_Param: 'Username'),
  'Password'=(Action_Param: 'Password'),
  'First_Name'=(Action_Param: 'First_Name'),
  'Last_Name'=(Action_Param: 'Last_Name'),
  'Phone_Number'=(Action_Param: 'Phone_Number')]
[/Inline]
```

Security Tips

This section provides some important tips and concepts about using security on a Lasso Professional Server server.

General

- Grant the least number of privileges necessary for a user to access the resources they need. By denying all accesses and only granting permissions when needed, unauthorized uses of the server can be minimized. Additional permissions are easy to grant within Site Administration.
- Turn off Classic Lasso support to prevent the ability for site visitors to submit Lasso commands directly to your Web server through URLs or HTML form submissions.
- Never allow untrusted parties to write Lasso code in Lasso pages stored on your server. All Lasso tags should be authored by trusted developers. By default only code which is stored within valid Lasso pages (or database fields) can be executed by Lasso. Consider carefully before allowing code from other sources to be executed.
- Use operating system privileges to protect sensitive files. The security of a Web site goes only as far as the security which protects the Lasso pages and databases files that comprise it. Create user accounts so that only trusted users can modify or read critical files.
- Store backed-up setup files in a secure location. These files contain all of the passwords and security settings for Lasso Professional Server.
- Use SSL security when appropriate. Site Administration and custom-built Web site administration interfaces can be accessed through SSL for greater security.
- Disable any tag modules which will not be in use. Remove any third-party LCAPI or LJAPI tag modules or data sources which are not in use. Remove any libraries or LassoApps from LassoStartup that are not in active use. Disable LJAPI if no LJAPI tag modules are in use.

Databases

- Never allow database files to be served directly by the Web server application. SQLite, FileMaker Pro, or other third party database files should never be stored within the Web server root. The database security in SQLite and FileMaker Pro is largely designed to prevent access to the databases through client applications or the graphical user interface. If a user gains access to the database file itself, they can usually read the data from it using a text editor.
- Store sensitive data encrypted within the database. The [Encrypt_...] tags can be used to encrypt data so only a Lasso page with the appropriate [Decrypt_...] tag can access it. Personal information and sensitive data like credit card numbers can be stored encrypted within the database, and only decrypted when necessary to process the order. Encryption tags are discussed in detail in the *Encoding* chapter of the Lasso 8.5 Language Guide.
- If sensitive information such as a record ID, password, or credit card number needs to be passed in a URL, or in a place where it can be viewed in the HTML source code, use of the [Encrypt_...] tags is recommended to encrypt the data so only a Lasso page with the appropriate [Decrypt_...] tag can access it.

User Authentication

- Once authentication information is entered into a Web browser for a specific Web server that same authentication information will be sent with each subsequent request.
- Usernames and passwords are cached by many Web browsers for easier authentication. If the username and password for the site administrator or any Lasso user is cached, then subsequent users of the Web browser may be able to access Site Administration or portions of a Web site without being prompted for authentication.
- If Site Administration or a protected area of a Web site is visited using a public browser such as in a library or Internet café, any requests to store cookies or cache usernames and passwords should be denied, and the history of the browser should be cleared before logging off of the machine.

- The site administrator is not bound by any of the settings established in Lasso Security. When a user is authenticated as the site administrator it may be possible to perform actions which are normally not allowed by Lasso Security.
- Sessions can be used to provide an alternate method of authentication for users. Sessions rely on cookies or information passed in URLs to authenticate a user. Sessions can be deleted explicitly, or will automatically timeout after a set period of inactivity.

Testing Solutions

- Each Web browser implements authentication in a slightly different fashion. If a Web site must be accessed from many different Web browsers, it should be tested on each one to ensure that the authentication works properly.
- Since authentication information is cached, the Web browser should be quit between each test of a different username and password. On some systems it might be necessary to quit all Internet applications in order to clear out cached usernames and passwords.
- Be careful with testing solutions using the same browser which has been used to access Site Administration. Since the site administrator can access any resources within Lasso, it can be difficult to tell how Lasso Security is set up if the current user is the site administrator.

15

Chapter 15

URL Rewrite

This chapter documents the URL Rewrite section in Site Administration, which allows the site administrator to set up rewrite rules that can perform automatic includes or redirects. This chapter describes the following:

- *Introduction* describes how Lasso's URL rewrite functionality works and compares it to Web server-based URL rewrite.
- *Rules, Triggers, and Actions* describes the triggers which are available to determine if a rule should run and the actions that can be performed by a rule.
- *Setting Up Rules* describes the interface in Site Administration which can be used to create and edit URL rewrite rules.

Introduction

URLs can reference any resource which can be served by a Web server, but in practice they are most often used to reference specific files. URL rewrite functionality makes it possible to use URLs for their original purpose and to sever the one-to-one correspondence between URLs and files.

For example, the following two URLs would traditionally reference a pair of files `rocket.html` and `station.html` which are located in the Web server root.

```
http://www.example.com/rocket.html  
http://www.example.com/station.html
```

However, in a database driven Web site each of these pages may depend primarily on content which is stored in a database. The files on disk might just be templates into which dynamic content is placed for the files are served to the site visitor. In this case, it makes sense to create a single `template.lasso` file which is used as the template for both files. The particular content which is to be served is passed in a URL parameter. The following two URLs might serve the same content as the URLs above.

```
http://www.example.com/template.lasso?page=rocket  
http://www.example.com/template.lasso?page=station
```

URL rewrite can be used to map the original URLs into these dynamic URLs. This allows the site visitor to bookmark the simple URL and for Lasso to serve the page effectively using the URL referencing the template and selecting the particular page using a URL parameter.

```
/rocket.html → /template.lasso?page=rocket
```

Web Server Rewrite

Most Web servers have built-in rewrite technology. Apache Web server includes the `mod_rewrite` module which allows URLs to be rewritten by rules defined in the Apache configuration file. WebSTAR V includes similar rewrite technology accessible through its administration interface. Although Microsoft IIS does not have rewrite technology built-in, third-party modules are available which provide this capability.

Deciding whether to use the Web servers built-in rewrite tools or Lasso's URL rewrite tools depends on several factors. Some of the advantages of each method are described below.

Web Server Rewrite Advantages

- The primary advantage of Web server based rewrite is that it can work on any URLs, even those which are not automatically processed by Lasso. For example, Web server based rewrite can more easily process rules about image files or extensionless files which are not normally processed by Lasso.
- There may be a performance benefit to using Web server based rewrite since the server can intervene earlier in the file serving process, before the file has even been handed off to Lasso for processing.

Lasso URL Rewrite Advantages

- Lasso's URL rewrite tools allow Lasso-specific actions which are not possible when using Web server based rewrite including using library or include files, setting Lasso variables, logging values, or triggering Lasso's built-in error processing.
- Lasso's URL rewrite technology is Web server independent. It allows the same rewrite rules to be used with a site no matter what Web server is being used to server the site.
- Lasso's URL rewrite tools use familiar Lasso concepts to designate rules, triggers, and actions. There is an easier learning curve for these tools than with some Web server based rewrite methodologies.

Note: Lasso's URL rewrite tools are a potential replacement for the "error.lasso" techniques which were used to trap file not found errors in earlier versions of Lasso.

Rules, Triggers, and Actions

Lasso's URL rewrite tools work by designating a series of rules. The rules are executed in order each time a page is processed by Lasso. The rules are processed prior to Lasso reading or parsing the designated Lasso page and can be used even for URLs which do not specify a literal Lasso page.

Each rule has a set of one or more triggers which determine if the rule is run and a set of one or more actions which are only run if the rule is triggered. Rules can be set up to run only if all triggers' criteria are met or if any one trigger's criterion is met. After a rule's actions are performed, either additional rules can be processed, or Lasso can be told to stop processing rewrite rules for that request.

Note: The URL rewrite rules are programmed not to modify requests for Lasso's built-in administration interfaces including `SiteAdmin.LassoApp`.

Triggers

Each trigger works like a comparison expression with a field, an operator, and a value. For example, a trigger to match any .lasso files would be `[Response_FilePath] Ends With ".lasso"`. The field `[Response_FilePath]` and the operator `Ends With` are selected from pop-up menus and the value `.lasso` is typed into a text input in the Site Administration interface.

Other trigger examples include:

- `[Response_FileExists] Is False` triggers if the requested file does not exist.
- `[Client_IP] Equals "192.168.0.*"` triggers if clients are within the specified IP range.
- `[Cookie: "UserID"] Does Not Equal ""` triggers if the cookie UserID is not empty.

- [Client_Header: "User-Agent"] Contains "Safari" triggers if the client is using the Safari browser.

The following table lists the triggers which are available for Lasso's URL rewrite rules.

Table 1: Trigger Fields

Trigger Field	Description
[Client_URL]	The full URL which was used to trigger the current page load including protocol, server name, directories, Lasso page name, and URL parameters. http://localhost/folder/file.lasso?name=value
[Server_Name]	The name of the server which was referenced by the current request.
[Server_Port]	The port which is being used to serve the current request. Either 80 for normal HTTP requests or 443 for HTTPS requests.
[Response_FilePath]	The path to the file referenced by the current request including directories and Lasso page name. Note that this field returns the path to the referenced file even if it does not exist.
[Response_FileExists]	True or false depending on whether the file referenced by [Response_FilePath] actually exists on the server or not. This field is usually used to redirect to a URL that does exist.
[Client_IP]	The IP address of the client.
[Client_Header]	References a header sent by the client with the current request. Requires the name of the header to be inspected (e.g. Content-Type, Accept-Encoding, or User-Agent).
[Cookie]	References a cookie sent by the client with the current request. Requires the name of the cookie to be inspected.
[Action_Param]	References any of the action parameters for the current request. Requires the name of the action parameter to be inspected. Note that the action parameters include both the URL and form parameters.
[Client_GetParam]	References any of the URL parameters for the current request. Requires the name of the URL parameter to be inspected.
[Client_PostParam]	References any of the form parameters for the current request. Requires the name of the form parameter to be inspected.
[Error_Code]	References the current set error code. Note that this cannot be used to detect -9984 file not found errors.
[Error_Msg]	References the current set error message.
[Variable]	References a variable. Requires the name of a variable.
[Global]	References a global variables. Requires the name of a global variable.
[True]	Allows a trivial trigger which always executes to be created.

The following table lists the available operators for use in URL rewrite triggers. These correspond closely to Lasso string comparison symbols and Lasso database search operators.

Table 2: Trigger Operators

Rule	Description
Is True	Matches if the trigger field returns true or is non-empty.
Is False	Matches if the trigger field returns false or is empty.
Equals	Matches if the trigger field matches the specified value. Case insensitive.
Not Equal To	Matches if the trigger field does not match the specified value. Case insensitive.
Contains	Matches if the trigger field contains the specified value. Case insensitive.
Does Not Contain	Matches if the trigger field does not contain the specified value. Case insensitive.
Begins With	Matches if the trigger field begins with the specified value. Case insensitive.
Ends With	Matches if the trigger field ends with the specified value. Case insensitive.

Matches Regular Expression	The value specifies a regular expression which is run against the trigger field. Matches if the regular expression matches any substring of the trigger field. Note also that the sub-patterns of the regular expression are stored and can be used within an action.
Does Not Match Regular Expression	The value specifies a regular expression which is run against the trigger field. Matches if the regular expression does not match any substring of the trigger field.
Less Than	Matches if the trigger field is less than the value. The comparison is either numeric or alphabetic.
Less Than or Equal	Matches if the trigger field is less than or equal to the value. The comparison is either numeric or alphabetic.
Greater Than	Matches if the trigger field is greater than the value. The comparison is either numeric or alphabetic.
Greater Than or Equal	Matches if the trigger field is greater than or equal to the value. The comparison is either numeric or alphabetic.

Most of the operators are used to compare the trigger field with a value. The Is True and Is False operators do not require values. The Matches Regular Expression and Does Not Match Regular Expression operators require a valid regular expression in the value. If the regular expressions have sub-patterns then the values of those sub-patterns can be retrieved using the [Trigger] token within the action. See below for examples.

Actions

The actions of a rule are performed if the triggers of the rule match their respective criteria. Actions can be used to redirect the site visitor to another URL, to include a file which is processed before the code in the Lasso page itself is processed, to set variables or global variables, or to serve an entirely different page than that referenced in the URL. Actions can be predefined or can use the values from the triggers as part of their parameters

Some action examples include:

- Redirect_URL: `http://www.example.com[Response_FilePath]` which redirects the client's Web browser to a URL based on the URL which they were trying to access, but on a different server.
- Redirect_URL: `/default.lasso?page=[Trigger: 1]` which redirects the client's Web browser to a specific Lasso page with the value of the first trigger as a parameter.
- Include: `/header.lasso` includes a file which is processed before any code in the file referenced by the URL request is processed.
- Rule actions can be chained together. The following five rules would set three variables to sub-patterns of the first trigger, include a file `onefile.lasso` and then abort. The trigger in this case might be of the form `[Response_FilePath] Matches Regular Expression "/(.*?)/(.*?)/(.*?)/.*"` which uses a regular expression with three sub-patterns to parse the incoming URL.

```
Variable: 'tab1' = "[Trigger: 1,1]"
Variable: 'tab2' = "[Trigger: 1,2]"
Variable: 'tab3' = "[Trigger: 1,3]"
Include: "/onefile.lasso"
Abort
```

The following table lists the actions which are available for Lasso's URL rewrite rules.

Table 3: Actions

Rule	Description
[Variable]	Sets a named variable to a value.
[Global]	Sets a named global variable to a value.
[Redirect_URL]	Redirects the client's Web browser to a specified URL.

[Include]	Includes a specified file which will be processed before any code in the URL specified by the client is processed. Note that if this action is used in concert with an [Abort] then the included file essentially takes the place of the file specified by the client since that file will never be processed.
[Library]	Like an include, but the results of the included file are not returned.
[Log_Always]	Logs a message to Lasso's console.
[Log_Critical]	Logs a critical error message.
[Log_Warning]	Logs a warning message.
[Log_Detail]	Logs a detail level message.
[Log_SQL]	Logs an action statement.
[Log_Deprecated]	Logs a deprecated functionality warning.
[Fail]	Fails with a specified error code and error message. The error code can be caught by a later URL rewrite rule or otherwise will trigger Lasso's built-in error processing.
[Abort]	Aborts processing of the current page. No further URL rewrite rules will be processed, but any at-end page handlers will be processed.

Many of the actions such as [Variable], [Redirect_URL], and the [Log...] tags require a value. The value is specified in the Site Administration interface. The values can include a number of tokens which are replaced with values from the triggers. These tokens work identically to the Lasso tags which they correspond to. All of the available tokens are listed in the following table.

Table 4: Action Tokens

Rule	Description
[Action_Param: "name"]	Returns the value of the specified action parameter.
[Client_GETParam: "name"]	Returns the value of the specified URL parameter.
[Client_POSTParam: "name"]	Returns the value of the specified form parameter.
[Cookie: "name"]	Returns the value of the specified cookie.
[Error_Code]	Returns the current error code.
[Error_Msg]	Returns the current error message.
[Response_Path]	Returns the path to the file requested by the client including the directories and the file name.
[Response_FilePath]	Returns the path to the file requested by the client including only the directories without the file name.
[Response_FileName]	Returns the name of the file requested by the client.
[Server_Name]	Returns the host name referenced by the client.
[Server_Port]	Returns the port accessed by the client. Either 80 for normal HTTP requests or 443 for HTTPS requests.
[Trigger: "Number"]	Returns the value for the specified trigger. This token can be used to get the value of any trigger no matter what operator was used within the trigger.
[Trigger: "Number", "Part"]	Returns a specific sub-pattern from a trigger which used a regular expression operator. The regular expression must use parentheses (...) to define at least one sub-pattern. The part should be omitted to return the entire trigger value.

Note: General LassoScript expressions cannot be used when specifying URL rewrite rules. The tokens in the table above must be written exactly as specified in order to be processed properly. Arbitrary code can be executed using an [Include] or [Library] action.

Setting Up Rules

The current rules can be seen in the *Setup > URL Rewrite > Rewrite Rules* section of Lasso Site Administration. Each rule is listed with its name, status, triggers, and actions. Selecting the name of a rule allows the rule to be edited. New rules can be added by selecting the Add Rule section (see below).

Figure 1: URL Rewrite Rules



The controls in the left sidebar allow the following actions.

- **Reload Rules** – Reloads the rules in the URL rewrite system. Rules which are added or edited are not made part of the live URL rewrite system until they are reloaded. This allows the administrator to edit a collection of rules as a batch and then make them all live at once.
- **Disable Rules** – Disables the URL rewrite system. This option can be useful when testing new rules to temporarily disable all of the URL rewrite rules.
- **Refresh** – Refreshes the display showing any rules which have been added.

Adding a Rule

A new rule can be added by selecting the *Setup > URL Rewrite > Add Rule* section. The initial screen allows one trigger and one action to be created for the rule. If the rule requires additional triggers or actions then they can be added after the rule has been created. Lasso will not make the new rule live until the URL rewrite system is reloaded. A new rule has the following characteristics:

- **Name** – The name of the rule, used strictly for display in the administration interface.
- **Status** – Allows rules to be activated or deactivated individually.
- **Perform Action If** – Allows rules to be set to perform their actions if all of their triggers' conditions are met or if any one trigger's condition is met. By default all of the triggers' conditions must be met.
- **Trigger/Action** – Lists a summary of the triggers and actions for the rule.
- **If Triggered** – By default all of the rules in the URL rewrite system will be run one by one. If this setting is set to Stop After Actions then the URL rewrite system will stop processing for the current URL after this rule performs its actions. This prevents later rules from further modifying the result of this rule.
- **Priority** – Rules are executed in order of their priority. Modifying the priority allows rules to be re-ordered. See below for details about the trigger and action sections.

Figure 2: Add Rule

Editing a Rule

When a rule is edited the *Setup > URL Rewrite > Edit Rule* section is shown. This screen has the same basic interface as the Add Rule section described above. The triggers and actions for the rule can be modified by selecting the summary of the trigger or action. The Add Trigger... and Add Rule... links allow new triggers or actions to be added. See below for details about editing or adding triggers or actions.

Figure 3: Edit Rule

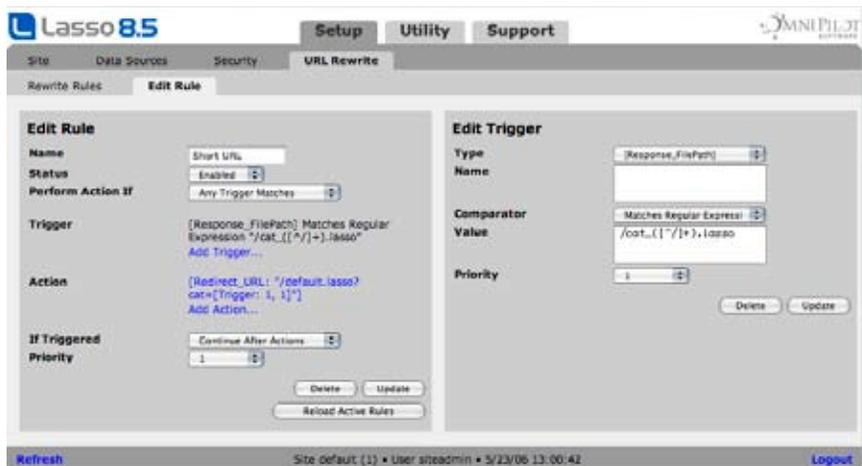
Triggers

This section describes the characteristics of triggers. The same basic interface is used when initially adding a rule, editing a trigger that is part of a rule, or adding a new trigger to a rule. A trigger takes the form of a comparison expression. See above for a full list of trigger types.

[Action_Param: 'myparam'] Contains "myvalue"

- **Type** – The type of trigger. Each type corresponds to a Lasso tag such as [Response_FilePath] or [Action_Param].
- **Name** – If the type requires a name value then it can be specified here. For example, [Action_Param] requires the name of the action parameter to test.
- **Comparator** – The comparator to use to check the value against the type. The comparators are equivalent to those which can be used when specifying database queries.
- **Value** – The value for the comparison. Most comparators require a string value. Is True or Is False do not require a value. The Greater Than and Less Than comparators are usually used with a numeric value. The Matches Regular Expression comparator requires a regular expression value
- **Priority** – Allows the trigger to be re-ordered. Triggers are performed in order and must be referenced by order in the [Trigger] token within an action.

Figure 4: Edit Trigger



Actions

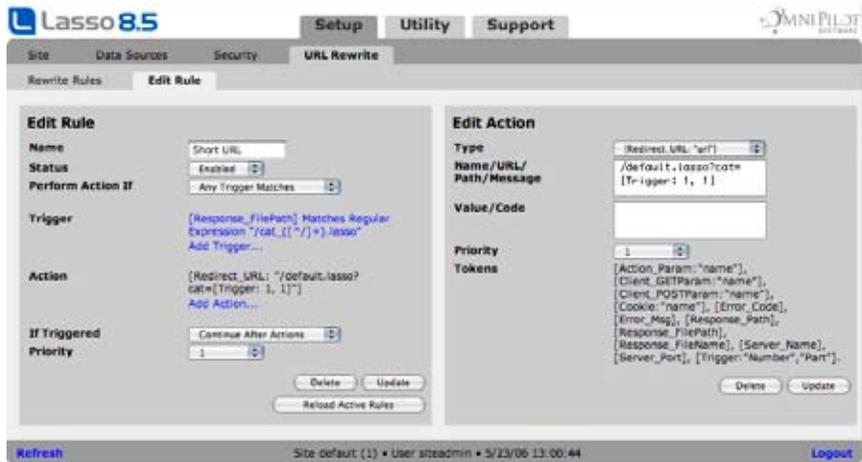
This section describes the characteristics of actions. The same basic interface is used when initially adding a rule, editing an action that is part of a rule, or adding a new action to a rule. An action takes the form of a simple Lasso tag. See above for a full list of action types and tokens.

[Redirect_URL] http://www.example.com[Response_FilePath]

- **Type** – The type of action. Each type corresponds to a Lasso tag such as [Redirect_URL] or [Log_Critical].
- **Name/URL/Path/Message** – If the type requires a parameter then it can be specified here. For example, [Redirect_URL] requires the URL to redirect to, and [Log_Critical] requires the message to log. The parameter value can use any of the token values to substitute in the result of various Lasso tags or the matched value from a trigger.
- **Value/Code** – If the type requires an additional parameter then it can be specified here. For example, [Fail] requires both a message and an error code and [Variable] requires a variable name and value. The parameter value can use any of the token values to substitute in the result of various Lasso tags or the matched value from a trigger.

- **Priority** – Allows the actions to be re-ordered. Actions are performed in order and the order can be important for certain sequences of actions. For example, one action can set a variable using the type [Variable] and then a later action can reference that value using the [Variable] token.

Figure 5: Edit Action



Note – Only the tokens which are listed can be used within the parameters of an action. The [Include] or [Library] types can be used to execute arbitrary code if necessary.

16

Chapter 16 Site Utilities

This chapter documents the *Utility* section in Site Administration, which provides the administrator with tools for managing and maintaining the Lasso Professional Server server. This chapter describes the following:

- *SQL Browser* describes how to issue SQL queries to configured databases in Site Administration.
- *Email* describes managing the email queue, sending a basic email message, and establishing email settings.
- *Events* introduces Lasso events, and describes how they can be set and managed.
- *Errors* describes viewing and setting preferences for the Lasso error log.
- *LassoApps* describes how to build and manage LassoApps on the Lasso Professional Server server.
- *Caches* describes how to view and manage developer-defined caches on the Lasso Professional Server server.

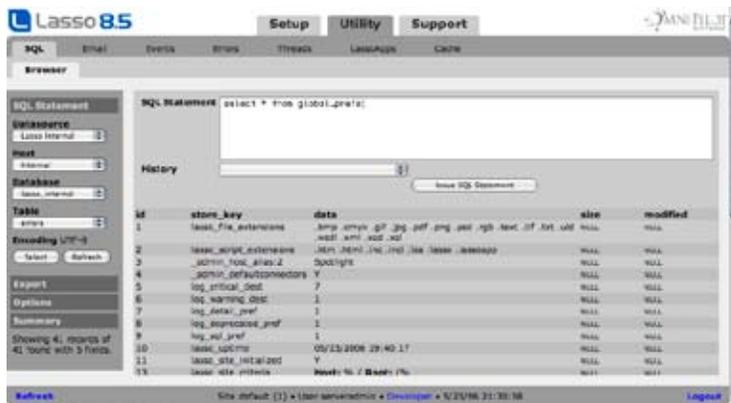
SQL Browser

The SQL section provides a Web-based interface that allows one to issue SQL queries to Lasso-accessible SQL databases. A Lasso-accessible SQL database is any SQLite, MySQL, or SQL-compliant JDBC database that has been set up and enabled in the *Setup > Data Sources* section of Site Administration.

Issuing SQL Statements

The SQL Browser page is where SQL statements can be entered. The results of the query will be presented at the bottom of the page with one column for each field that is returned. The results could contain actual field data from a database search, or could contain synthetic field data which is generated by the database in order to format the results of various SQL statements and functions. See the examples that follow for more information.

Figure 1: SQL Browser Page



To issue a SQL statement:

- 1 Select a data source connector from the Connector pull-down menu. Only connectors to data sources that support SQL will be shown.
- 2 Select a data source host from the Host pull-down menu.
- 3 Select a database to issue a SQL statement from the Database pull-down menu. Only databases enabled in the *Setup > Data Sources* section will be shown.
- 4 Enter a SQL query in the SQL Statement field.
- 5 Select the Issue SQL Statement button. The results of the statement are shown in the lower panel, which by default shows No Results Found.

SQL Statement History

A special feature of the SQL Browser is that the last ten statements entered into the SQL Statement field will be displayed in a pull-down menu that appears to the right of History. In order to issue a statement stored in the History pull-down menu, simply select the statement from the pull-down menu, and it will automatically appear in the SQL Query field. Then select Issue SQL Query.

SQL Export

Selecting the Export area in the sidebar allows the results of a SELECT statement to be exported. Setting one or more options and then selecting Export Results will result in a text file being sent to the browser containing the current result set.

- **Format** – Specifies what format the export should be in. Options include Tab Delimited, Comma Delimited, SQL Inserts, or XML Data.
- **Field Names** – Specifies whether field names should be included as the first line of tab or comma delimited files. For XML data this preference controls whether field names are used as the tag names surrounding each field value or if a generic field container is used.
- **Line Endings** – Sets the line endings for the export to `\n`, `\r`, or `\r\n`.

SQL Options

Selecting the Options area in the sidebar allows the following options to be set:

- **Show Fields** – Allows the maximum number of fields displayed in the results to be selected. The options are 4 Fields, 8 Fields, 12 Fields, 16 Fields, and All Fields. If more than eight fields are selected, then the results panel will expand to the right to accommodate the extra fields.

- **Max Records** – Allows the maximum number of records displayed in the results to be selected. The options are 10 Records, 25 Records, 50 Records, 75 Records, 100 Records, and All Records.
- **Max Field Lines** – Allows the maximum number of lines of text to be displayed per field in the results to be selected. One line consists of approximately 20 characters horizontally. The options are 1 Lines, 4 Lines, 7 Lines, 10 Lines, and All Lines. Selecting All Lines will display the entire contents of the field.
- **Submit on Return** – If this option is set then a return in the SQL Statement field will submit the statement.

Email

The Email section allows the administrator to manage the site Lasso email queue, send email, and set site email options in Lasso Professional Server. Full documentation for sending an email in Lasso Professional Server can be found in the *Email* chapter in the Lasso 8.5 Language Guide.

The email queue can be monitored in several different ways. This section provides an overview of the tabs in the *Utility > Email* section, followed by more in depth information about each sub-section.

- **Email Queue** – Provides an overview of messages which are currently being sent by Lasso. Each message in the queue corresponds to one call to the [Email_Send] tag. If a message is addressed to users on different SMTP servers then a message may be split into several parts. Information about one part and one error are included inline within each message display.
- **Email Parts** – Provides an overview of message parts which are currently being sent by Lasso. A message which is addressed to users on different SMTP servers may be split into several parts which are sent independently. Once all of the parts of a message have been sent, the message status is updated. This section allows the parts that currently being sent to be viewed. Either all parts or the parts for a single message are shown.
- **Email Errors** – The email sending system records errors for each message and part which is being sent. The errors section allows all errors or the errors for a single message to be shown.
- **Email Routes** – Allows the SMTP routes which have been looked up to be viewed and modified.
- **Email Send** – Allows an email to be sent through Lasso for testing.
- **Setup** – The preferences for the email sending system can be configured in this section.

Email Queue

The email queue logs all email messages that are sent from within Site Administration or via the [Email_Send] tag. Messages remain in the queue while they are being sent to the SMTP mail server looked up by Lasso or specified in the [Email_Send] tag by the developer. For more information, see the *Email* chapter in the Lasso 8.5 Language Guide.

Selecting the Stop Queue temporarily suspends email from sending. This allows the administrator to inspect the state of the email queue without messages sending. The button then changes to Start Queue, and can be re-selected to start email sending again. Emails are sent when the server is restarted even if the email queue is stopped.

Each message may include one or more parts depending on the recipients of the message. A message to one recipient always has one part. If a message has several recipients who are in different domains then the message will be split into one part per domain. Information about the first part of a message is shown inline within the message display. The message display also includes the most recent error returned for the message.

Messages in the queue can be viewed by status by selecting a message category from the pull-down menu in top panel, and then selecting the Select button. These categories are as listed below.

- **All Messages** – Shows all messages in the email queue.
- **Queued Messages** – Shows messages currently queued in the email queue.
- **Sent Messages** – Shows all messages that have been sent previously in the email queue.
- **Message Errors** – Shows all messages that were unable to be sent in the email queue due to an error.

Figure 2: Email Queue Page



The following actions can be performed on a single message or on multiple messages using the controls to the left.

- **Try Now** – Available for queued messages. Instructs Lasso to attempt to send the message now without waiting for the retry delay to pass.
- **Requeue** – Available for messages which have encountered an error. Instructs Lasso to queue the message again from scratch. Messages should be requeued if there was an issue with the local SMTP server or after modifying Lasso's email settings to clear up an error.
- **Hold** – Holds a message indefinitely.
- **Resume** – Resumes sending a held message.
- **Delete** – Deletes a message, permanently removing it from the queue. If the message has multiple parts they will be deleted as well.
- **Show All** – Shows all parts or errors for a message.
- **Clear** – Clears the errors for a message.

Email Parts

This section lists all of the message parts which Lasso currently has in its queue. The same controls as shown above for the email queue also work in this section. By default the parts of all messages are shown, but if a Show All link is followed from a message part in the email queue then only parts for that message are shown.

Email Errors

This section lists all of the errors which the email sending system has encountered. It works similarly to the *Utility > Errors* section which shows general Lasso errors. By default the errors encountered while sending all messages are shown, but if a Show All link is followed from a message error in the email or parts queues then only errors for that message are shown.

Email Routes

When Lasso sends an email message it first looks up the SMTP server for the recipient of the message using a domain name server. The address of that SMTP server is stored as the route for the domain. The email routes section can be used to view the Cached Routes which Lasso has looked up.

The User Routes allow custom routes to be specified for particular domains. For example, the local domain may need to route to an internal SMTP server rather than the Internet-facing server. The email routes section allows user routes to be created for these special cases. A user route includes the following fields.

- **Domain** – The domain must correspond to the domain which will be seen in the recipients of email messages. Email to joe@example.com has a domain of example.com.
- **Host** – The host name or IP address of the SMTP server to send messages for the domain to.
- **Port** – The port to use on the SMTP server. Leave blank to use the default.
- **Username** – The username to use for SMTP authentication on the SMTP server. This is usually left blank since authentication is not normally required for sending messages to a domain's target SMTP server.
- **Password** – The password to use for SMTP authentication.
- **Retry Delay** – The retry delay for messages to this SMTP server in seconds.
- **Timeout** – The timeout for connections to this SMTP server in seconds.
- **Encryption** – If encryption is enabled then SSL/TLS will be used to encrypt all communication to the SMTP server. This setting should be used only when needed since the encryption incurs significant overhead and may slow down mail operations.
- **Comment** – Allows a comment about the route to be recorded so the administrator can later remember why it was added.

Figure 3: Email Routes Page



Sending Email

The Send Email page allows email messages to be sent. After the message is queued, one returns to the Email Queue page. While Lasso Professional Server provides robust support for custom email sending programmed within your Lasso pages, the Send Mail form allows the administrator to quickly test email sending.

Figure 4: Send Email Page



To send an email:

- 1 Enter the email host name in the SMTP Host field and the SMTP Port and SMTP AUTH Username and Password if required.
- 2 Enter the recipient's email address in the To field.
- 3 Enter the sender's email address in the From field.
- 4 Enter the subject of the email in the Subject field.
- 5 Enter the body of the email in the Body field.
- 6 Enter the email address of any carbon-copy recipients in the CC field (optional).
- 7 Enter the email address of any blind carbon-copy recipients in the BCC field (optional).
- 8 Select the Send Message button. The message will be inserted into the email queue database and sent out per settings specified on the Email Setup page.

Email Setup

The Setup page allows one to set preferences for the [Email_Send] tag. This includes setting the administrator email address, default host IP address, message retention, and system automation.

Figure 4: Setup Page



The following describes the options in the Send Email page:

- **Send Directly to SMTP Servers** – If set to yes then Lasso will attempt to send messages directly to the recipients SMTP server.
- **Ignore [Email_Send] -Host Params** – If set to yes then Lasso will always send messages based on the preferences set on this page and will ignore -Host, -Port, -Username, -Password, and -Timeout parameters specified in the [Email_Send] tag.
- **Default SMTP Host** – The default SMTP server for sending email. If this is specified then the Host field in the Send Email page and the -Host parameter in the [Send_Email] tag are optional.
- **Default SMTP Port** – An alternate port to connect to on the SMTP host. This will usually be set to the default of 25.
- **SMTP Host Timeout** – The number of seconds Lasso should wait before timing out the remote host. This will usually be set to the default of 15 seconds.
- **SMTP AUTH Username and Password** – Optional username and password to use for SMTP AUTH authentication on the default SMTP host.
- **Maximum Retries** – The maximum number of times a message send will be attempted (5 attempts by default).

- **Email Retry Delay** – The number of seconds to pause between retrying a message send (30 seconds by default).
- **Email Sweep Delay** – The amount of time the email sender sleeps before it checks to see if there are any messages to send (15 seconds by default).
- **Retain Successful Messages** – Retains details about messages that have been successfully sent in the email queue (No by default).
- **Retain Error Messages** – Retains messages which have an error when sending (Yes by default).
- **Retain Encoded Messages in Log** – Retains the full MIME encoded messages using the options above. Otherwise, only the headers are retained (No by default).
- **Maximum # Queues** – The maximum number of outgoing SMTP connections which should be made (5 by default).
- **Send # Messages Per Connection** – The maximum number of messages that should be sent to a single server in one connection (100 by default).
- **Override Lasso Service IP** – The IP address which should be sent to remote SMTP servers. If blank (the default) the actual IP address of the machine hosting Lasso Service will be used. This should only be set to a different value for multi-homed servers or if serving from behind a NAT service or firewall.

Email settings can be changed by entering new values for the options above and selecting the Update button. Selecting the Stop Mail Queue button stops the email queue while site options are being set.

Events

The Events section allows the administrator to list events, get the status of an event, clear the event queue, or delete events that have been scheduled. Additional documentation on events and the [Event_Schedule] tag can be found in the *Control Tags* chapter in the Lasso Professional Server Language Guide.

Introduction to Events

Events in Lasso Professional Server take the form of URL requests. For user-defined URL requests, the event scheduler can operate as a virtual site visitor which visits specific URLs at specific times. The event scheduler can be used to schedule page loads on the local server or even on remote servers.

Normally, the pages which the event scheduler has been programmed to visit will be pages which perform specific tasks that only the event scheduler has access to. For example, a folder named Events could be created, and all pages that the event scheduler calls could be placed in it. A file `SendEmail.lasso` would then be scheduled as:

```
http://www.example.com/Events/SendEmail.lasso
```

Events can be scheduled to execute at a specific time, but the actual time the event is executed may vary based on server traffic and other conditions. Events can only be certain to execute within one minute of their scheduled time.

Event Queue

The Event Queue page provides a list of scheduled events. The events can be filtered by type, and events can be deleted if they have not yet been performed.

All events that have been scheduled and not performed yet are shown in the queue. If an event repeats over time, then it will remain in the queue until the administrator-defined period for that event has ended. Once an event has ended, it is removed from the queue.

Figure 5: Event Queue Page



The last error for each event is shown inline within the event display. Each event also supports the following actions.

- **Pause** – Temporarily pauses an event.
- **Resume** – Resumes executing a paused event.
- **Edit** – Allows the event to be edited so the URL or frequency of a recurring event can be adjusted. Events are automatically paused when they are being edited and resumes once the editing is complete.
- **Delete** – Permanently removes the event from the queue.

Selecting the Pause All Events temporarily suspends all queued events from being executed. This allows the administrator to delete events that are causing problems or schedule a batch of events. The button then changes to Resume Events, and can be re-selected to start event execution again. Events are restarted when the server is restarted even if the event queue is stopped.

The panel shows all queued events for the selected queue that have not been performed yet. The URL, time of next run, start and end dates (for repeating events), options, and username are shown for each event. The administrator may delete an event from the current queue by selecting the Delete link under Actions for that event.

Selecting the Delete All Events button in the sidebar removes all shown events from the queue shown.

Event Errors

This section lists all of the errors which the event scheduling system has encountered. It works similarly to the *Utility > Errors* section which shows general Lasso errors. By default the errors encountered while executing all events are shown, but if a Show All link is followed from an event in the event queue then only errors for that event are shown.

Scheduling Events

The Schedule Event page provides a convenient interface for scheduling events. Events can also be scheduled programmatically via the [Event_Schedule] tag, as discussed in the *Control Tags* chapter in the Lasso 8.5 Language Guide. All scheduled events can be viewed in the Event Queue page.

Figure 6: Schedule Event Page



The Perform Event pull-down menu specifies how often an event should be run. The remaining fields in the Schedule Event page will vary depending on which option is selected for Perform Event, and the options are as follows:

- **Once** – Event is performed once at a specified date and time.
- **Weekly** – Event is repeated once a week on a specified day and time between the Start Date/Time and End Date/Time.
- **Daily** – Event is repeated once a week on a specified day and time between the Start Date/Time and End Date/Time.
- **Custom Repeat** – Event is repeated at a specified time interval (in minutes) between the Start Date/Time and End Date/Time.

Because the input fields vary for each option, scheduling an event to be performed once, weekly, daily, and in a custom repeating fashion are described in separate procedures below.

To schedule an event to be performed once:

- 1 Enter the URL address of the event in the Event URL field.
- 2 Enter the specific date and time on which the event will occur in the Event Date/Time field (mm/dd/yyyy hh:mm:ss format).
- 3 Select Yes or No from the pull-down menu in the Continue After Restart field. If set to Yes, the event will continue after the Web server is stopped and restarted.
- 4 Enter any username required to perform the event in the Username field (optional). An event will require a username if the page to be loaded is protected by Lasso Security, and requires a username for access. For more information on usernames, see the *Setting Up Security* chapter.
- 5 Enter any password required to perform the event in the Password field (optional). The same conditions apply to passwords as for usernames.
- 6 Select Schedule Event.

To schedule a weekly event:

- 1 Enter the URL address of the event in the Event URL field.
- 2 Select a day of the week on which the event will be performed from the Day of Week pull-down menu (e.g. Sunday, Monday, ...).
- 3 Enter the time of day on which the event will be performed from the Time of Day pull-down menu (24-hour hh:mm:ss format).
- 4 Enter the start date of the event in the Start Date/Time field. The date on which a repeating event is scheduled to start. The format is mm/dd/yyyy hh:mm:ss. If the start date is specified without a time, the time will default to 00:00:00 (12:00 AM).
- 5 Enter the ending date of the event in the End Date/Time field. The date when a repeating event is scheduled to end. The format is the same as Start Date/Time.
- 6 Select Yes or No from the pull-down menu in the Continue After Restart field. If set to Yes, the event will continue after the Web server is stopped and restarted.
- 7 Enter any username required to perform the event in the Username field (optional). An event will require a username if the page to be loaded is protected by Lasso Security, and requires a username for access. For more information on usernames, see the *Setting Up Security* chapter.
- 8 Enter any password required to perform the event in the Password field (optional). The same conditions apply to passwords as for usernames.
- 9 Select Schedule Event.

To schedule a daily event:

- 1 Enter the URL address of the event in the Event URL field.

- 2 Enter the time of day on which the event will be performed from the Time of Day pull-down menu (24-hour hh:mm:ss format).
- 3 Enter the start date of the event in the Start Date/Time field. The date on which a repeating event is scheduled to start. The format is mm/dd/yyyy hh:mm:ss. If the start date is specified without a time, the time will default to 00:00:00 (12:00 AM).
- 4 Enter the ending date of the event in the End Date/Time field. The date when a repeating event is scheduled to end. The format is the same as Start Date/Time.
- 5 Select Yes or No from the pull-down menu in the Continue After Restart field. If set to Yes, the event will continue after the Web server is stopped and restarted.
- 6 Enter any username required to perform the event in the Username field (optional). An event will require a username if the page to be loaded is protected by Lasso Security, and requires a username for access. For more information on usernames, see the *Setting Up Security* chapter.
- 7 Enter any password required to perform the event in the Password field (optional). The same conditions apply to passwords as for usernames.
- 8 Select Schedule Event.

To schedule a custom repeating event:

- 1 Enter the URL address of the event in the Event URL field.
- 2 Enter the number of minutes at the beginning of the start date after which you want the event to begin in the Repeat Delay field. Repeat Delay is the interval (in minutes) by which an event will repeat.
- 3 Enter the start date of the event in the Start Date/Time field. The date on which a repeating event is scheduled to start. The format is mm/dd/yyyy hh:mm:ss. If the start date is specified without a time, the time will default to 00:00:00 (12:00 AM).
- 4 Enter the ending date of the event in the End Date/Time field. The date when a repeating event is scheduled to end. The format is the same as Start Date/Time.
- 5 Select Yes or No from the pull-down menu in the Continue After Restart field. If set to Yes, the event will continue after the Web server is stopped and restarted.
- 6 Enter any username required to perform the event in the Username field (optional). An event will require a username if the page to be loaded is protected by Lasso Security, and requires a username for access. For more information on usernames, see the *Setting Up Security* chapter.
- 7 Enter any password required to perform the event in the Password field (optional). The same conditions apply to passwords as for usernames.
- 8 Select Schedule Event.

Errors

The Errors section allows the administrator to view and delete errors, warnings detail messages, deprecated functionality warnings, and action statements that have been logged both internally in Lasso Professional Server and via the [Log] tag, as well as set site logging options. For more information on the [Log] tag, see the *Files and Logging* chapter in the Lasso 8.5 Language Guide.

Important: Configuring error logging in Site Administration is not the same thing as configuring page-level error handling, such as for syntax errors and security errors. Page-level error handling is described in the *Error Reporting* chapter of the Lasso 8.5 Language Guide. The default Lasso error page can be configured in the *Setup > Site > Syntax* section of Site Administration.

Lasso Errors

The Lasso Errors page lists all errors and system messages stored in the Lasso_Internal database. Errors and system messages can also be deleted within this interface.

Figure 7: Errors Page



The Lasso Errors page contains a list of all error messages received by their date, time, level, and message. The list is navigable via the Prev and Next buttons at the bottom of the panel.

Filtering Messages

The Lasso Errors pull-down menu allows messages to be filtered by level, which includes Critical Errors, Warnings, Details, Action Statements, and Deprecated. Each error level is described individually in the following *Error Log Setup* section. Individual Lasso page and database errors are not logged here, but are rather controlled using the [Error...] tags. For more information, see the *Error Control* chapter in the Lasso 8.5 Language Guide.

Deleting Messages

Selecting the Delete Errors Shown button deletes only the errors shown in the current panel, while selecting the Delete All Errors button deletes all errors in the current selected category. A confirmation dialog box will be displayed upon selecting either of these buttons.

Error Log Setup

The Setup page allows the administrator to set site logging options for Lasso Professional Server. These logging options are not limited to errors, and any feedback that occurs within the Lasso Service console can be logged for display in the Lasso Errors page.

Figure 8: Error Setup Page



The Error Setup panel allows the administrator to set unique log routing for the five log levels:

- **Critical Errors** – This includes fatal errors that affect the overall functionality of Lasso Professional Server. This can include startup errors, module and connector errors, and system-related errors that cause Lasso Professional Server not to function properly.

- **Warnings** – This includes non-fatal errors that do not affect the overall functionality of Lasso Professional Server, but the administrator should be aware of. This can include non-fatal startup errors, duplicate module or connector errors, SQL status errors, and logging errors.
- **Details** – This logs all processes performed by Lasso Service, which are not limited to errors.
- **Deprecated** – This logs warnings when deprecated tags or other functionality are used. Any deprecated tags or functionality may not be supported in a future version of Lasso.
- **Action Statements** – This logs the action statements (e.g. SQL statements) from inline database actions.

For each feedback type, the administrator can choose to log or display feedback in any of the following three places:

- **Lasso Service Console** – If selected, feedback will be displayed in the Lasso Service console window when Lasso Service has been started as an application. For more information on starting Lasso Service as an application, see the configuration chapters of this guide.
- **Error Database** – If selected, feedback will be logged to the `_Errors` table in the `Lasso_Internal` database and displayed in the Lasso Errors page in Site Administration.
Note: The `_Errors` table will store a maximum of 10,000 records. Once this limit has been reached, the oldest records will automatically be deleted.
- **LassoErrors.txt** – If selected, feedback will be logged to the `LassoErrors.txt` file in the Lasso Professional Server folder in the hard drive.

Options are changed by checking or unchecking the desired values in the Error Setup panel and selecting the Update button. The Reset button returns all options to their default values. Selecting the Refresh button reloads the Setup page with its currently stored values.

Note: For information on how to programmatically log custom data to the logs described above, see the *Files and Logging* chapter in the Lasso 8.5 Language Guide.

LassoApps

The *Utility > LassoApps* section in Site Administration allows administrators to create and manage LassoApps. The following is an introduction to LassoApps, and describes how they can be used.

Introduction to LassoApps

LassoApps (short for “Lasso Applications”) allow entire Lasso solutions—including Lasso pages, html pages, images, and any other files—to be compiled into a single file called a LassoApp. Once a LassoApp has been compiled, the original Lasso code can no longer be viewed (client-side or server-side). LassoApps are served from a Web server using Lasso Professional Server, and are capable of the same functionality as traditional Lasso pages.

Important – Although it is difficult to extract the original LassoScript from a LassoApp it is possible to see the strings that are referenced within the LassoApp. LassoApps should be used to protect your LassoScript code, but are not a recommended method of storing passwords or other sensitive data.

LassoApp Advantages

LassoApps are useful for hiding implementation details, locking Lasso pages, selling locked solutions, and installing pre-compiled solutions. Just like regular Lasso pages, LassoApps are portable and cross-platform compatible. LassoApps also offer the following advantages over regular Lasso pages.

- **Performance** – Performance is enhanced over regular Lasso pages due to the file existing as a single, cohesive unit. Once a LassoApp is read by Lasso once, it is cached in the RAM memory of the Web server and is no longer accessed from the hard disk until it is dumped from memory or the system is restarted.

- **Size** – LassoApps take up less disk space than the sum of its Lasso pages since the overhead associated with multiple files (file headers, image files, etc.) is reduced.
- **Security** – The code within a LassoApp is stored securely into a single file and cannot be viewed in a text or HTML editor. There is no way to extract Lasso pages from a LassoApp.
- **Portability** – LassoApps are easily portable since they consist of only one file, and all file paths to the different parts of the solution are retained internally.

LassoApp Characteristics

LassoApp file names end with the .LassoApp extension, instead of .html or .lasso. The ability to serve LassoApps must be enabled within Lasso Professional Server site settings in order to serve LassoApps. If LassoApp serving is not enabled in Site Administration, then the user will receive a security error.

The functionality within LassoApps is only effective if the functions being performed by the LassoApp are permitted by Site Administration. LassoApp developers have the ability to create routines for checking whether or not the necessary permissions and settings are enabled, and instruct the administrator on what to do to configure Lasso Professional Server settings for use with specific LassoApps. For further information on LassoApps, see the *LassoApps* chapter in the Lasso 8.5 Language Guide.

Building LassoApps

LassoApps can be created by the Lasso site administrator in the *Utility > LassoApps > Build* section of Site Administration. Creating a LassoApp involves supplying three pieces of information:

- The LassoApp root folder, which is the folder that contains all Lasso pages, HTML pages, images, and any other files to be compiled.
- The LassoApp root file, which is the default page or start page for the Lasso solution to be compiled.
- The file name for the compiled LassoApp.

Preparing source files for LassoApp Builder

Before building a LassoApp, all links to files within the LassoApp root folder must be replaced with the [LassoApp_Link] Lasso tag. The [LassoApp_Link] Lasso tag is what allows internal navigation between pages in a LassoApp. Links to files outside of the LassoApp root folder do not need to be formatted with this tag. Below are examples of how to express links within Lasso pages contained within a LassoApp.

- Links to files within the LassoApp root folder:

```
<a href="page.html">Link</a>
```

```
→ <a href="[LassoApp_Link:'page.html']">Link</a>
```

- Forms within the LassoApp root folder:

```
<Form Action="page.html"> ... </Form>
```

```
→ <Form Action="[LassoApp_Link:'page.html']"> ... </Form>
```

- Images within the LassoApp root folder:

```

```

```
→ 
```

All links to files within the LassoApp root folder need to be formatted as shown above before using LassoApp Builder. Not doing this will prevent the LassoApp from being created. For more information on preparing Lasso pages to be converted into a LassoApp, see the *LassoApps* chapter in the Lasso 8.5 Language Guide.

Note: Adding these tags in regular Lasso pages will not affect how the links, forms, and images function. Lasso will ignore these tags until they are compiled into a LassoApp using LassoApp Builder.

Using LassoApp Builder

The *Utility > LassoApps > Build* section of Site Administration is where LassoApps can be compiled.

Figure 9: LassoApp Build Page



To create a LassoApp:

- 1 Format all links to files within the LassoApp source files using the [LassoApp_Link] tag as shown previously.
- 2 The folder containing the LassoApp should be located within your Web server root. Note the URL that would be used to access these LassoApp files. For example:

`http://www.example.com/myLassoApp/default.lasso`

- 3 Go to the *Utility > LassoApps > Build* section of Site Administration.
- 4 Select Custom Root... from the LassoApp Folder pop-up menu. Specify the path to the folder containing your LassoApp from the root of the Web server. The for the example URL shown above the path would be:

`/myLassoApp/`

- 5 Enter the name of the default page for the solution in the Entry File field. The Entry File is the first page the user will see when accessing the LassoApp. For the example URL shown above the entry file would be:

`default.lasso`

- 6 Select Create LassoApp. This will create a LassoApp with the same name as the folder that contains the LassoApp files within your Web server root. The LassoApp created using the instructions above would be located at this URL:

`http://www.example.com/myLassoApp.LassoApp`

Alternately, select Download LassoApp. This will create the LassoApp with the same name as the folder that contains the LassoApp files within your Web server root. The LassoApp is then downloaded by your Web browser and can be moved from the download location into the Web server root.

For more information on creating LassoApps, see the *LassoApps* chapter. in the Lasso 8.5 Language Guide.

Note: It is also possible to create LassoApps by placing the LassoApp folder in a BuildLassoApps folder within the LassoAdmin folder of the current site. The current site must have been granted permission to access this folder by the server administrator. For the default site the full path to this folder would be:

`.../Lasso Professional 8/LassoSites/1-default/LassoAdmin/BuildLassoApps/`

Viewing and Dumping LassoApps

When a LassoApp is first called in a Web browser, it gets cached in the computers's RAM for efficient serving. If a new version of a particular LassoApp is loaded to the Web server, then the LassoApp will need to be removed from the system cache before the new version can be accessed.

The *Utility > LassoApps > List* section is where the administrator can view the names of LassoApps that have been loaded on the server, and dump them from the system cache.

Figure 10: List LassoApps Page



To remove a LassoApp from the system memory cache:

In the LassoApp Cache panel, select the minus (-) button next to the name of the LassoApp (e.g. MyLassoApp.LassoApp) to dump from memory. This panel contains all LassoApps that have been served by Lasso Professional Server since startup.

Selecting Dump All LassoApps will remove all LassoApps from the system cache. After LassoApps have been dumped from the system cache, new versions of the LassoApps may be loaded.

LassoApp Settings

The *Utility > LassoApps > Setup* section is where the administrator can enable or disable LassoApps in Lasso Professional Server.

Figure 11: LassoApp Settings Page



Disabling LassoApps prevents the execution of any custom or third party LassoApp not included with Lasso Professional Server. The LassoApps that cannot be disabled are Admin.LassoApp, GroupAdmin.LassoApp, LassoScriptReference.LassoApp, and Setup.LassoApp in the LassoStartup folder.

To enable or disable LassoApps:

- 1 In the Settings panel, select Enabled or Disabled from the LassoApps pull down menu.
- 2 Select the Update button.

Caches

New caching tags in Lasso Professional Server make it possible for developers to cache content in their pages using the [Cache] ... [/Cache] tags. These custom caches are stored in memory on the server. These caching tags allow developers to reduce database and server load by having Lasso only recalculate various portions of a page periodically.

The Lasso site administrator has global control over all caches stored on the Lasso Professional Server server. The *Utility > Cache* section of Site Administration provides information about all current caches, allows caches to be reset, and allows preferences for the caching mechanism to be set.

Viewing and Dumping Caches

The *Utility > Cache > List* page in Site Administration provides a list of active caches on the server, including cache name, the size of cached data, and the efficiency of the cache. This page also provides total cache usage statistics since Lasso was started (cache statistics always reset when Lasso Service is restarted).

Figure 12: Cache List Page



The Cache List panel displays a list of all caches that have been served by Lasso Professional Server since startup, the size and data type of cached data, and an efficiency score. A cache's efficiency score is the number of times the cache has been loaded minus the number of times it has been refreshed or expired, then divided by the number of loads (loads - stores / loads). Clicking the Refresh button updates the panel with the latest information.

Clicking on the name of a cache shows the Cache Detail panel to the right, where statistics about the cache may be viewed, and the cache can be deleted or refreshed. Refreshing a cache allows cache statistics to persist, whereas deleting a cache does not.

To refresh a cache:

- 1 Select the name of the cache to reset from memory in the Cache List panel. This panel contains the names of all caches that have been loaded on the Lasso Professional Server server.
- 2 In the Cache Detail panel, Select the Expire Now button. Once a cache is manually expired, its contents will be refreshed the next time it is loaded.

To delete a cache:

- 1 Select the name of the cache to delete from memory in the Cache List panel.
- 2 In the Cache Detail panel, Select the Delete button. Once a cache is deleted, it is completely removed from the system memory (including the statistics for the cache) until it is loaded again.

Cache Settings

The *Utility > Cache > List* page in Site Administration allows site cache preferences to be set. These preferences apply to all caches loaded on the server, and will override any preferences set by developers.

Figure 13: Cache Settings



The following cache preferences may be set:

- **Allow Caching** – Globally enables or disables the [Cache_...] tags on the Lasso Professional Server server.
- **Maximum Expiration** – A global option for the maximum length of time (in minutes) that any cached content can be stored before it is automatically reset.
- **Maximum Cache Size** – Specifies an upper limit on the amount of data (in kilobytes) that can be stored in a cache. Defaults to 32.
- **Host Specific Caches** – Specified whether or not caches may be accessed across different host names (as output by the [Server_Name] tag).

Chapter 17

Building and Browsing Databases

This chapter documents the Lasso Database Browser, which is a LassoApp included with Lasso Professional Server for interacting with all Lasso-configured databases and building custom MySQL databases.

- *Overview* describes the features of the Lasso Database browser.
- *Database Browser* describes how to search, add, update, view, and delete records in Lasso-enabled databases, and how to import and export data.
- *Database Builder* describes how the administrator can create databases, tables, fields, and indices within SQLite and MySQL data sources.

Overview

The Lasso Database Browser allows administrators to interact with the records of all databases enabled in Site Administration. This includes searching, viewing, adding, updating, and deleting records in any database accessible by Lasso. The Lasso Database Browser also allows the administrator to build custom MySQL databases using a Web-based interface for use in Lasso solutions.

Accessing the Lasso Database Browser

The Lasso Database Browser consists of the DatabaseBrowser.LassoApp file located in the LassoApps folder of the Lasso Professional Server application folder. The Lasso Database Browser is a LassoApp file, which is accessed via a Web browser similar to a standard HTML page.

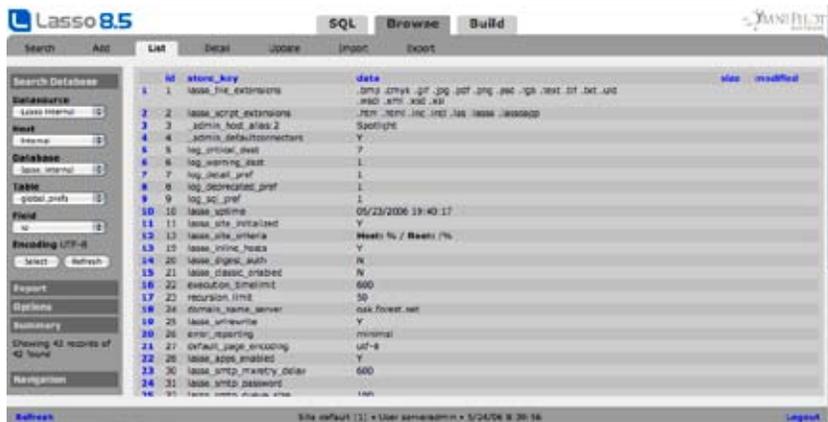
The Lasso Database Browser can be accessed in any Web browser that meets the Web browser requirements listed in the configuration chapters in this guide.

To access the Lasso Database Browser:

In a Web browser, visit <http://www.example.com/Lasso/DatabaseBrowser.LassoApp>. Replace `www.example.com` with your domain name, IP address, or `127.0.0.1` if on a local machine. If an error is displayed, make sure Lasso Service is running as described in the configuration chapters of this guide.

The database browser for a specific host can be visited by loading Site Administration for that host and then using the link in the Support section.

Figure 1: Lasso Database Browser



Security Note: The Lasso Database Browser will prompt for the Lasso site administrator username and password. Only the site administrator may use the Lasso Database Browser.

Lasso Database Browser Features

The Browse section allows the administrator to test databases once they have been enabled, as well as modify records in existing databases. It is always recommend that databases be tested in the Browse section of the Lasso Database Browser before attempting to interact with them via custom Lasso code. The Browse section also allows the administrator to import and export data.

The Build section allows the administrator to create databases, tables, fields, and indices, and edit the schema of SQLite and MySQL database via a Web-based interface. No knowledge of SQL is required to build databases using the Build section.

Database Browser

The Database section of the Lasso Database Browser allows administrators to interact with database records and includes the following pages:

- **Search** – Allows the administrator to search records in a selected table.
- **Add** – Allows the administrator to add records to a selected table.
- **List** – Lists records within a selected table.
- **Detail** – Displays the contents of a selected record.
- **Update** – Allows the administrator to update or delete a record within a selected table.
- **Import** – Allows the administrator to import a text data file to a selected table.
- **Export** – Allows the administrator to export data from a selected table to a text format.

Selecting a Database to Work With

The sidebar is where the data source connector, host, database, table, and key field of the records to be browsed are first specified. The administrator can use the sidebar on any page within the browse section.

Figure 2: Settings Sidebar**To select a connector, host, database, table, and key field:**

- 1 In the sidebar, select a data source connector from the Datasource pull-down menu. When a pull-down menu option is selected, the Select page will automatically update to show the correct information for that option in the pull-down menus below.
- 2 Select a data source host from the Host pull-down menu.
- 3 Select a database from the Database pull-down menu.
- 4 Select a table from the Table pull-down menu.
- 5 Select the Key Field from the Key Field pull-down menu. The key field of a table is typically the ID field, and is used to store values unique to each record.
- 6 Select the Select button.

Searching Records

The Search page allows the administrator to search records in a selected table. All fields are shown by default for the selected table, and both Search and Find All buttons return results in the List page. The data type of each field is displayed to the right of each field by default. For more information on data types, consult the appropriate third-party documentation for the data source being used.

Figure 3: Search Page**To search records in a table:**

- 1 Enter the information you wish to search for in the appropriate field next to each field name. If data is entered in more than one field, the found set will only contain records containing all values searched for.
- 2 Select Search Records. If records matching the search parameters are found, the administrator will be taken to the List page to view the found set. If no matching records are found, a No Records Found message will be displayed in the Search page.

The sidebar displays the current database, table, and key field being used in the Database section. These can be changed by selecting new values in each pull-down menu, and then selecting the Select button. Selecting a new database or table will automatically change the fields in the Search page to represent the new table.

Selecting the Show Options button displays the Field View Options panel to the right with the first field selected. Here, the administrator may set field display options for each field.

Search Options

Selecting the name of a field in the Search Records panel displays the Field View Options panel to the right for that field.

Figure 4: Search Options



The following input and display options can be edited for each field.

- **Field** – Displays the name of the selected field in a pull-down menu. Selecting a different field will display the options for that field. Select a field here if the desired field is not currently visible in the Search Records panel.
- **Type** – Displays the data type of the field. This is useful for determining an appropriate input type.
- **Display on Page** – Selecting No causes the field to no longer be displayed in the Search Records panel.
- **Input Type** – Designates what HTML input type will be used for the field in the Search page. Options are as follows:

Automatic – Determines the input type based on the data type of the field. For example, a VARCHAR or INT type would be a text input, a TEXT data type would be a text area input, and an ENUM data type would be a pull-down menu.

Text Input – Uses the HTML input type Text for the field.

Password Input – Uses the HTML input type Password for the field.

Text Area – Uses the HTML input type Textarea for the field.

Select List – Uses the HTML input type List for the field.

Pull-Down Menu – Uses the HTML input type Select for the field.

Check Boxes – Uses the HTML input type Check for the field.

Radio Buttons – Uses the HTML input type Radio for the field.

- **Input Size** – Determines the horizontal and/or vertical size of the field input. This option appears if the Text Input, Password Input, Select List, or Pull-Down Menu options are selected, and the options are Small, Medium, and Large.
- **Position** – Allows a custom field display order to be set. Fields assigned a lower number will be displayed ahead of fields assigned a higher value. Once the Position value has been changed, the Position value of all other fields will adjust to accommodate for the new field value.
- **Display Field Type** – Selecting No causes the field data type to no longer appear to the right of each field in the Search Records panel.

Options are changed by selecting the desired values from the pull-down menus and selecting the Apply button. The Reset button returns all options to their default values. Selecting the Hide Options button hides the Field View Options panel.

Adding Records

The Add page allows the administrator to add a record to the selected table. The Add Record panel displays all fields within the table by default, and displays an input field to the right of each field name. The input type used for each field (text input, pull-down menu, radio buttons, etc.) can be customized in the Field View Options panel, which is described later in this section.

Figure 5: Add Page



To add a record to table:

- 1 Enter the information you wish to add in the appropriate text field next to each field name. The data type of each field is displayed to the right by default.
- 2 Select Add Record. The results of an added record are displayed in the Detail page.

The sidebar displays the current database, table, and key field being used in the Databases section. The key field of the current record may be changed by selecting a new key field from the Key Field pull-down menu, and then selecting the Select button. The fields shown in the Add page will change to reflect the new table if a new database or table is selected.

Selecting the Show Options button displays the Field View Options panel to the right with the first field selected. Here, the administrator may set field display options for each field.

Add Options

Selecting the name of a field in the Add Record panel displays the Field View Options panel to the right for that field.

Figure 6: Add Options



The following input and display options can be edited for each field.

- **Fields** – Displays the name of the selected field in a pull-down menu. Selecting a different field will display the options for that field. Select a field here if the desired field is not currently visible in the Add Record panel.
- **Type** – Displays the data type of the field. This is useful for determining an appropriate input type.
- **Display on Page** – Selecting No causes the field to no longer be displayed in the Add Record panel.

The bottom right panel contains navigation controls allowing the administrator to view different sets of records from the found set. The Next and Previous buttons take the administrator to the next and previous sets in the found set. The First and Last buttons will take the administrator to the first and last record in the found set. The Jump button allows the administrator to specify the record number from the found set in which to begin the list with.

The bottom left panel contains sort options that allow the administrator to control how the records are sorted in the List page. The Sort Field pull-down menu contains the field by which the records are sorted, and the Sort Order pull-down menu specifies whether the records are sorted in an ascending (A-Z, 1-9) or descending (Z-A, 9-1) order. The sort options can be changed by selecting a value from each pull-down menu and selecting the Sort button.

Export

Selecting the Export area in the sidebar allows the current results to be exported. Setting one or more options and then selecting Export Results will result in a text file being sent to the browser containing the current result set.

- **Format** – Specifies what format the export should be in. Options include Tab Delimited, Comma Delimited, SQL Inserts, or XML Data.
- **Field Names** – Specifies whether field names should be included as the first line of tab or comma delimited files. For XML data this preference controls whether field names are used as the tag names surrounding each field value or if a generic field container is used.
- **Line Endings** – Sets the line endings for the export to `\n`, `\r`, or `\r\n`.

List Options

Selecting the name of a field displays the Field View Options panel in the lower left panel for that field. Doing so will limit the number of records shown in the List page to one record while display options are set.

Figure 8: List Options



The following input and display options can be edited for each field.

- **Field** – Displays the name of the selected field in a pull-down menu. Selecting a different field will display the options for the different field. Select a field here if the desired field is not currently visible in the List page.
- **Type** – Displays the data type of the field.
- **Display on Page** – Selecting No causes the field to no longer be displayed in the List page. In the List page, this only applies to fields in positions 1-8 (see Position below).
- **Display Size** – Allows the maximum number of lines of data displayed for the selected field to be set. Options are 1 Line, 4 Lines, 7 Lines, 10 Lines, and All Lines.

- **Position** – Allows a custom field display order to be set. In the List page, only the first eight fields will be displayed. Fields assigned a lower number will be displayed ahead of fields assigned a higher value. Once the Position value has been changed, the Position value of all other fields will adjust to accommodate for the new field value.
- **List Records** – Allows the number of records displayed in the List page to be set. This can be 5, 10, 25, 50, or 100.
- **List Response** – Determines where the administrator will be taken after a record is selected in the List page. Detail takes the administrator to the Detail page when a record is selected. Update takes the administrator to the Update page when a record is selected. No Details Links disallows records from being selected in the List page.

Options are changed by selecting the desired values from the pull-down menus and selecting the Apply button. The Reset button returns all options to their default values. Selecting the Hide Options button hides the Field View Options panel and returns the administrator to the regular List view with all records displayed.

Displaying Record Details

The Detail page displays the field data for the selected record. All fields in the record are displayed by default.

Figure 9: Detail Page



Record Detail

The Record Detail panel shows the field data for the record. The data type of each field is displayed to the right of the data for that field. Selecting the Update button will take the administrator to the Update page, while selecting Delete button will display a confirmation dialog asking the administrator to confirm the delete. Once this is done, the administrator will be returned to the List page. If a record is deleted, it cannot be retrieved.

The sidebar displays the current database, table, and key field being used in the Databases section. The key field of the current record may be changed by selecting a new key field from the Key Field pull-down menu, and then selecting the Select button. Since the database or table of the current record displayed cannot be changed, selecting a new database or table will return the administrator to the Select page.

The bottom panel contains navigation controls allowing the administrator to instantly switch to another record in the found set to be displayed in the Detail page. The Next and Previous buttons take the administrator to the next and previous records in the found set. The First and Last buttons will take the administrator to the first and last record in the found set. The Jump button allows the administrator to specify the record number from the search results of the record to be displayed.

Selecting the Show Options button displays the Field View Options panel to the right with the first field selected. Here, the administrator may set field display options for each field.

Detail Options

Selecting the name of a field in the Record Detail panel displays the Field View Options panel to the right for that field.

Figure 10: Detail Options



The following input and display options can be edited for each field.

- **Field** – Displays the name of the selected field in a pull-down menu. Selecting a different field will display the options for the different field. Select a field here if the desired field is not currently visible in the Record Detail panel.
- **Type** – Displays the data type of the field. This is useful for determining an appropriate input type for the field.
- **Display on Page** – Selecting No causes the field to no longer be displayed in the Record Detail panel.
- **Display Size** – Changes the number of lines of data that are displayed for the selected field. Options are 1 Line, 4 Lines, 7 Lines, 10 Lines, and All Lines.
- **Position** – Allows a custom field display order to be set. Fields assigned a lower number will be displayed ahead of fields assigned a higher value. Once the Position value has been changed, the Position value of all other fields will adjust to accommodate for the new field value.
- **Display Field Types** – Selecting No causes the field data type to no longer appear next to each field in the Record Detail panel.

Options are changed by selecting the desired values from the pull-down menus and selecting the Apply button. The Reset button returns all options to their default values. Selecting the Hide Options button hides the Field View Options panel.

Updating and Deleting Records

The Update page allows the administrator to update or delete records within a database, and displays all fields and their current values for the selected record by default.

- **Field** – Displays the name of the selected field in a pull-down menu. Selecting a different field will display the options for the different field. Select a field here if the desired field is not currently visible in the Update Record panel.
- **Type** – Displays the data type of the field. This is useful for determining an appropriate input type for the field.
- **Display on Page** – Selecting No causes the field to no longer be displayed in the Update Record panel.
- **Input Type** – Designates what HTML input type will be used for the field in the Update page. Options are as follows:

Automatic – Determines the input type based on the data type of the field. For example, a VARCHAR or INT type would be a text input, a TEXT data type would be a text area input, and an ENUM data type would be a pull-down menu.

Text Input – Uses the HTML input type Text for the field.

Password Input – Uses the HTML input type Password for the field.

Text Area – Uses the HTML input type Textarea for the field.

Select List – Uses the HTML input type List for the field. Any values defined in an ENUM or SET data type or FileMaker Pro value list for the field will automatically be displayed as selectable values.

Pull-Down Menu – Uses the HTML input type Select for the field. Any values defined in an ENUM or SET data type or FileMaker Pro value list for the field will automatically be displayed as selectable values.

Check Boxes – Uses the HTML input type Check for the field. Any values defined in an ENUM or SET data type or FileMaker Pro value list for the field will automatically be displayed as selectable values.

Radio Buttons – Uses the HTML input type Radio for the field. Any values defined in an ENUM or SET data type or FileMaker Pro value list for the field will automatically be displayed as selectable values.

- **Input Size** – Determines the horizontal and/or vertical size of the field input. This option appears if the Text Input, Password Input, Select List, or Pull-Down Menu options are selected from Input Type, and the options are Small, Medium, and Large.
- **Position** – Allows a custom field display order to be set. Fields assigned a lower number will be displayed ahead of fields assigned a higher value. Once the Position value has been changed, the Position value of all other fields will adjust to accommodate for the new field value.
- **Display Field Types** – Selecting No causes the field data type to no longer appear to the right of each field in the Update Record panel.

Options are changed by selecting the desired values from the pull-down menus and selecting the Apply button. The Reset button returns all options to their default values. Selecting the Hide Options button hides the Field View Options panel.

Importing Records

The Import page allows data to be imported to the selected table from a text file in a comma-delimited, tab-delimited, or XML format. Files can be uploaded from the local machine or can be placed in the Lasso Professional Server/LassoAdmin/Browse folder. Uploaded files are placed in this folder and then imported. This is the same folder which the Export section uses.

Note: The data in import files is always imported in the same order as the fields in the database. There is no option to match fields or import data out of order.

Figure 13: Import Page



The sidebar displays the current database, table, and key field being used in the Databases section. These can be changed by selecting new values in each pull-down menu, and then selecting the Select button. Selecting a new database, table, and keyfield will change the table to which data is imported.

Import File

The Import File panel allows the administrator to upload or select the file to be imported. These options are described below:

- **File Name** – Allows the administrator to select the text file to be imported. The pull-down menu lists all files currently stored in the Lasso Professional Server/lassoAdmin/Browse folder. The administrator must select a file and select the Apply button before the remaining options can be used.

Alternately, the file selector can be used to choose a file on the same machine as the Web browser. When the Apply button is selected the file will be uploaded into the Lasso Professional Server/lassoAdmin/Browse folder.

Import Settings

The Import Settings panel allows the administrator to set import options and see a preview of the data to be imported. This panel is only available when a file is selected in the Import File panel. The options available are described below:

- **Type** – Allows the data format of the text file to be explicitly specified. One of three formats may be selected:
 - Automatic – Site Administration automatically determines which format the text file is in and displays it in parentheses after Automatic. This is the default setting.
 - Comma Delimited – Specifies that the file to be imported is in comma delimited format.
 - Tab Delimited – Specifies that the file to be imported is in tab delimited format.
 - XML – Specifies that the file to be imported is in XML format.
- **Line Endings** – Allows the administrator to specify which line ending characters are used in the text file. Supported line ending characters vary from operating system to operating system, and the options are as follows:
 - Automatic – Site Administration automatically determines which line endings are used in the text file is in and displays it in parentheses after Automatic. This is the default setting.
 - UNIX/Mac OS X (*/n*) – Text file uses the */n* line ending, used in UNIX and Mac OS X operating systems.
 - Mac OS 9 (*/r*) – Text file uses the */r* line ending, used in the Mac OS 9 operating system.
 - Windows (*/r/n*) – Text file uses the */r/n* line ending, used in Windows operating systems.
- **Replace Matching Key Values** – Selecting Yes causes any data imported to the key field of the current table to be overwritten with new values. For example, if the key field is the ID field in a MySQL table, selecting Yes will cause whatever field that is matched with the ID field in the current database to be overwritten with new ID values, as if it were a new table and records are being added for the first time.

- **Skip First Record** – Selecting Yes does not import the first line of the file, which sometimes includes the names of each field and does not contain record data.
- **Import Preview** – Shows the first five lines of the text file to be imported. This is useful in that it shows the first line of the file to be imported, allowing the administrator to determine if the data and format of the file are correct.

Options are changed by selecting the desired values from the pull-down menus, selecting or deselecting a check box to the left of the desired options, and selecting the Apply button. The Reset button returns all options to their default values.

To import data to the current table:

- 1 In the Import File panel, select a data file to import from the File Name pull-down menu or upload a file from the local machine. Select the Apply button if no other options are shown.
- 2 In the Import Settings panel, select the data type of the text file from the Type pull-down menu, or verify that the Automatic type is correct by looking at the Import Preview field.
- 3 Select the line endings used in the text file in the Line Endings pull-down menu, or verify that the Automatic type is correct by looking at the Import Preview field. Knowing which operating system the file was created in is helpful for determining which line ending characters are used.
- 4 Select Yes for Replace Matching Key Values if you want the imported data to receive new key values in the current table. This is strongly recommended if data is being imported to a table with existing records, however, the field that is matched with the current key field will be overwritten.
- 5 Select Yes for Skip First Record if the first record in the text file contains field names as opposed to values. See the Import Preview field to determine this.
- 6 Select the Apply button.
- 7 Select the Perform Import button.

Exporting Records

The Export page allows the administrator to export data from a table to a text file in a comma-delimited, tab-delimited, or XML format. This feature allows the records in the current found set to be instantly exported to a text format that can be interpreted by any number of other data sources.

Files can be downloaded to the local machine or can be exported to the Lasso Professional Server/lassoAdmin/Browse folder. Downloaded files are first written into this folder and then streamed to the Web client. This is the same folder which the Import section uses.

Figure 14: Export Page



The sidebar displays the current database, table, and key field being used in the Databases section. These can be changed by selecting new values in each pull-down menu, and then selecting the Select button. Selecting a new database or table will automatically find all records in the new table, which will be the found set that is exported.

Export Settings

The Export Settings panel allows the administrator to set the format options in which the current found set will be exported. The Type pull-down menu allows the administrator to select from one of three data formats to export to. The remaining options (which the exception of Line Endings) will vary depending on which format is selected.

- **Type** – Allows the text format to be set for the data which will be exported.
 - Comma Delimited – Exports data to text in a comma-delimited format.
 - Tab Delimited – Exports data to text in a tab-delimited format.
 - XML – Exports data to text in an XML format.
- **Export Field Names** – Selecting Yes includes the names of each field to be included in the first line of data, similar to a spreadsheet header. Available for comma-delimited and tab-delimited formats.
- **Root Tag Name (XML Only)** – Allows the root XML tag name for the data file to be specified. All data in the text file will be contained in root tags.
- **Record Tag Name (XML Only)** – Allows the record XML tag name for the data file to be specified. All records will be contained in record tags.
- **Field Tag Name (XML Only)** – Allows the root XML tag name for the data file to be specified. All fields will be contained in field tags.
- **Line Endings** – Allows the administrator to specify which line ending character will be used in the text file. Supported line ending characters vary from operating system to operating system, and the options are as follows:
 - UNIX/Mac OS X (/n) – Uses the /n line ending, used in UNIX and Mac OS X operating systems.
 - Mac OS 9 (/r) – Uses the /r line ending, used in the Mac OS 9 operating system.
 - Windows (/r/n) – Uses the /r/n line ending, used in Windows operating systems.

Options are changed by selecting the desired values from the pull-down menus, selecting or deselecting a check box to the left of the desired options, and selecting the Apply button. The Reset button returns all options to their default values.

Export Preview

The Export Preview page displays the first five records of what the exported data will look like in the text file. To preview all data to be exported, select the Preview Full Export button.

The Perform Export button exports data to the Lasso Professional Server/lassoAdmin/Browse folder. The Download Export button opens a new browser window and automatically downloads the data.

To export data:

- 1 In the Export Format panel, select the data format to be exported to from the Type pull-down menu.
- 2 Select options for the specified format below the Type pull-down menu.
- 3 Select a line ending option from the Line Endings pull-down menu.
- 4 Select Apply.
- 5 Examine the data to be exported in the Export Preview panel to ensure it is the correct format needed for the data source that will import the file.
- 6 Enter a name for the text file to be created in the File Name field. To have the file name end with the date and time of creation, select Yes from the Append Date to File Name pull-down menu.
- 7 Select the Perform Export button or the Download Export button.

Note: The Export page will only export the current found set from the List page. To export all records in a table, make sure the Find All Records button was selected previously in the Select or Search page.

SQL Browser

The **Browse > SQL** section of the Lasso Database Browser provides a Web-based interface that allows the administrator to issue SQL queries to Lasso-accessible SQL databases. This includes any SQLite, MySQL, or SQL-compliant JDBC database that has been set up and enabled in the **Setup > Data Sources** section of Site Administration.

Figure 15: SQL Browser



The functionality of the **Browse > SQL** section is similar to that of the **Utility > SQL** section of Site Administration. For documentation on issuing SQL statements, see the *Site Administration Utilities* chapter.

Database Builder

The Build section of the Lasso Database Browser allows the administrator to create databases, tables, fields, and indices, and edit the schema of SQLite and MySQL database via a Web-based interface. This is useful for easily creating new MySQL databases for custom solutions, or for editing the schema (tables names, field names and types) of existing SQLite or MySQL databases.

Creating MySQL Databases

The Databases page is where the administrator can view all databases within a selected host and create new databases. The Databases Listing panel shows a listing of all databases within the selected host and data source connector.

Figure 16: Databases Page



The current data source connector and host can be changed using the Connector and Host pull-down menus. The administrator can view database details by selecting the database name, which will show the details to the right in the Database Detail panel.

Creating Databases

Selecting the Create Database button displays the Create Database panel to the right. The Create Database page is where the administrator can create new databases within a specified data source host.

Figure 17: Create Database Panel



To create a new database:

- 1 Enter the name of your database in the Name field.
Note: Per naming conventions in SQLite and MySQL, database names may only consist of alpha-numeric characters plus the `_` and `$` characters. For more information on naming conventions, see the `datasource` documentation.
- 2 Select Enabled or Disabled from the Status pull-down menu to enable or disable the database within Lasso.
- 3 Select Create Database.

Duplicating Databases

Selecting the Duplicate Database button shows the Duplicate Database panel to the right. The Duplicate Database panel is where the administrator can duplicate an existing database within a specified data source host. The administrator has the option to duplicate a database's schema only (tables and fields), or the schema along with all data (records).

Figure 18: Duplicate Database Panel



To duplicate an existing database:

- 1 Select the data source connector of the source database from the Connector pull-down menu. This can be either Lasso Connector for SQLite or Lasso Connector for MySQL.
- 2 Select the data source host of the source database from the Host pull-down menu.
- 3 Select the source database from the Database pull-down menu.
- 4 Select Yes from the Copy Data pull-down menu to copy all records from the source database to the new database. Selecting No copies only the database schema (table and field structure) with no records.
- 5 Enter the name of the new database in the Name field.
Note: Per naming conventions in SQLite and MySQL, database names may only consist of alpha-numeric characters plus the `_` and `$` characters. For more information on naming conventions, see the `datasource` documentation.
- 6 Select Enabled or Disabled from the Status pull-down menu to enable or disable the database within Lasso.
- 7 Select Duplicate Database. The new database will appear in the Database Listing panel to the left.

Updating and Deleting Databases

The administrator may update or delete a database by selecting the Update Database or Delete Database button in the Database Detail panel. To update a database, the administrator can change the name, alias, and status for the database via the Name, Alias, and Status fields, and select the Update Database button. Selecting the Delete Database button will display a confirmation message for deleting the selected database, and selecting OK will permanently delete the database.

Creating Tables

The Tables page is where the administrator can view all tables and create new tables within the selected database. The Table Listing panel shows the current database and lists all tables within the database. The administrator can view table details by selecting a table name, which will show details to the right in the Table Detail panel.

Figure 19: Tables Page



Creating Tables

Selecting the Create Table button shows the Create Table panel to the right. The Create Table page is where the administrator can create tables in the selected database.

Figure 20: Create Table Panel



To create a table in the selected database:

- 1 Enter the name of your table in the Table Name field.

Note: Per naming conventions in SQLite and MySQL, table names may only consist of alpha-numeric characters plus the `_` and `$` characters. For more information on naming conventions, see the datasource documentation.

- 2 Select Enabled or Disabled from the Status pull-down menu to enable or disable the table within Lasso.
- 3 Select Create Table.

The ID field is automatically created in each new table, which increments automatically with every new record added. There can only be one ID field per table.

Duplicating Tables

Selecting the Duplicate Table button shows the Duplicate Table panel to the right. The Duplicate Table panel is where the administrator can duplicate an existing table within a specified database.

Figure 21: Duplicate Table Panel



To duplicate a table in the selected database:

- 1 Select the data source connector of the source table from the Connector pull-down menu.
- 2 Select the data source host of the source table from the Host pull-down menu.
- 3 Select the database of the source table from the Database pull-down menu.
- 4 Select the source table from the Table pull-down menu.
- 5 Select Yes from the Copy Data pull-down menu to copy all records from the source table to the new database. Selecting No copies only the table schema (field structure) with no records.
- 6 Enter the name of the new table in the Name field. This new table will contain the information being duplicated, and will be created within the database currently selected in the Table Listing panel.

Note: Per naming conventions in SQLite and MySQL, table names may only consist of alpha-numeric characters plus the `_` and `$` characters. For more information on naming conventions, see the datasource documentation.

- 7 Select Enabled or Disabled from the Status pull-down menu to enable or disable the table within Lasso.
- 8 Select Duplicate Table. The new table will appear in the Tables Listing panel to the left.

Updating and Deleting Tables

The administrator may update or delete a table by selecting the Update Table or Delete Table button in the Table Detail panel. To update a table, the administrator can change the name, alias, and status for the table via the Name, Alias, and Status fields, and select the Update Table button. Selecting the Delete Table button will display a confirmation message for deleting the selected table, and selecting OK will permanently delete the table from the selected database.

Creating Fields

The Fields page is where the administrator can view all fields within the selected table and database. The Fields Listing panel includes a list of fields for the current table including all their editable attributes and the option to delete fields.

The SQL statement that will be used to modify the table can be viewed by selecting the Preview Change button. The modifications can be applied to the table by selecting the Alter Table button.

In addition to modifying the fields within the table, one field can be added by specifying its attributes in the New line at the end of the table. If more fields need to be added the Create Fields section can be used.

Figure 22: Fields Page



Each field includes the following details.

- **Name** – The name of the field.
- **Type** – The type of the field. Since SQLite is typeless the field type is only used to determine sort order (alphabetic or numeric).
- **Display** – An optional parameter which depends on the data type. In MySQL the display value is often required. See the MySQL documentation for more information.
- **Default** – An optional default value for the field.
- **Options** – The options for a field depend on the data source. The following options are commonly available.

Not Null – If checked then the field cannot contain *NULL* values.

Primary Key – One field in each table can be designated the primary key for the table.

Unique – Available on new fields controls whether the field should be constrained to only contain unique values.

Delete – This option deletes the field and all its contents.

Creating Fields

The Create Fields panel allows multiple fields to be created all at once. This section allows up to ten fields to be created at once. The SQL statement that will be used to modify the table can be viewed by selecting the Preview Change button. The modifications can be applied to the table by selecting the Alter Table button.

Figure 23: Create Field Panel



The options for each field are the same as those for modifying fields in the Field section.

Creating Indices

The database browser allows the indices of SQLite and MySQL tables to be manipulated.

An index in a MySQL database is an ordered list of the contents of a field or group of fields in a table (also known as the MYI file in the MySQL database folder). Indices in MySQL databases allow faster searches to be performed for the fields specified, and up to 25 indices can be specified for a MySQL table. For more information on MySQL indices, see the MySQL documentation.

The Indices page allows to administrator to view the current indices for a selected table, as well as create new indices that contain one or more fields.

Figure 24: Indices Page



The Index Listing panel shows a list of current indices in the current table. The administrator can view details for an index by selecting the index name, which will show the details in the Index Detail panel to the right.

Creating Indices

Selecting the Create Index button show the Create Database panel to the right. The Create Index panel is where the administrator can create new indices for the selected table. Unlike setting the index options for individual fields in the Fields page, indices created in the Index Detail panel may contain more than one field. A single index may contain up to 16 unique fields. To learn about the benefits associated with multiple field indexing, see the datasource documentation.

Figure 25: Create Index Panel

**To create a new index:**

- 1 Type a name for the index.
- 2 Select the fields that will comprise the index. In MySQL the entire length of CHAR or VARCHAR fields will be index and the first 255 characters of TEXT fields.
- 3 Select the type of index. The following types are available.
 - Normal – The default index type. Available in SQLite and MySQL.
 - Unique – The index will constrain the fields to contain only unique values. Available in SQLite or MySQL.
 - Full Text – A full-text index. Available only in MySQL.
 - Spatial – A spatial search index. Available only in MySQL.
- 4 Select the Create button The new index should appear in the Index Listing panel to the left.

Note: Primary key fields can be established in the *Fields* section.

Index Detail

The Index Detail panel shows all information available from the database for the selecting index. The following information is displayed:

- **Name** – The name of the index.
- **Field** – The names of the fields that the index contains. Multiple fields are delineated as Field 1, Field 2, etc.
- **Type** – The index type. For a list of possible types, see the next section.

The administrator may delete an index by selecting the Delete button in the Index Detail panel. Indices cannot be edited, only created and deleted. A confirmation message will always be displayed before an index is deleted.

A

Appendix A Glossary

This glossary is provided as a single location where the most critical terms which are used in the manual are defined. Please browse through the glossary to familiarize yourself with the terminology of Lasso 8. The glossary is divided into sections and the terms are listed alphabetically within each section.

User Types

Administrator – A person who has privileges to access the configuration and security settings of your server. Includes one server administrator and one site administrator per site.

Client – An application which access your Web site. A Web browser is the most common example of a client.

Developer – A person who creates Lasso pages. Developers include both HTML programmers and Lasso programmers.

User – A visitor to your site defined in Lasso Security or in the security system of a database. A visitor may authenticate themselves as a user in order to gain permission to execute restricted tags or database actions.

Visitor – An end-user who visits your site through a Web browser or other means. When a developer is testing a Web site, the developer is considered a visitor of the Web site.

Product Line

Lasso 8 – The name of the Lasso 8 product line including Lasso Professional Server and Lasso Studio.

Lasso Professional Server – The Lasso 8 product used to serve data-driven Web sites. Lasso Professional Server is commonly abbreviated LP8.

Lasso Studio – A Lasso 8 product used to build data-driven Web sites to be served by Lasso Professional Server. Lasso Studio installs as a set of extensions for Adobe GoLive or Macromedia Dreamweaver.

LassoScript 8 – LassoScript stands for Lasso Dynamic Markup Language, the name of the programming language implemented and supported by the Lasso 8 product line.

Architecture

Default Site – All incoming Web requests are handled by this site by default. The default site also serves to catch any incoming Web requests that do not match another sites' site criteria.

Lasso Apps – Compiled Lasso solutions. An entire folder of Lasso pages, HTML files, and images are compiled into a single LassoApp. LassoApps can be served on-demand from the LassoApps folder in the Lasso Professional Server application folder or from the Web server root.

Lasso Connector – A module or plug-in that provides connectivity between Lasso Service and an external application.

Lasso Data Source Connector – A module that installs into Lasso Service and provides connectivity to an external data source application. Built-in data source connectors are provided for MySQL, JDBC data sources, and FileMaker Pro. Third parties can build their own Lasso data source connectors using LCAPI, LJAPI, or Lasso tags.

Lasso Libraries – Collections of tags in the same namespace which are loaded on-demand from the LassoLibraries folder of the Lasso Professional Server application folder.

Lasso Modules – Extensions to Lasso that are installed into Lasso Service. There are two types of Lasso Modules: Lasso data source connector Modules and Lasso Tag Modules. Lasso Modules can be implemented in C/C++ through the Lasso C Application Programming Interface (LCAPI), or in Java through the Lasso Java Application Programming Interface (LJAPI).

Lasso Service – The core executable of Lasso Professional Server. It is implemented as a service application which can be started and stopped independently from the Web server. Lasso Service is the Lasso Web Data Engine.

Lasso Tag Module – A LassoSoft or third party provided module that installs into Lasso Service and implements one or more Lasso tags providing extended functionality.

Lasso Web Server Connector – A module or plug-in that installs into a Web Server and provides connectivity to Lasso Service. There are three initial Lasso Web server connectors: Lasso Connector for Apache, Lasso Connector for IIS, and Lasso Connector for WebSTAR V. Other connectors may be created using Lasso Connector Protocol.

Master Site – The master site routes incoming Web requests to each site based on their site criteria. The master site also serves the initialization pages and Server Administration interface.

Site – An independent Lasso Service process. Each site has its own modules, data sources, privileges, security settings, etc. Each site can be stopped and restarted without affecting other sites. Site criteria are used to determine which site will process incoming Web requests.

SQLite – The internal SQLite database provided with Lasso Professional Server. SQLite is required to store Lasso 8 settings and preferences, and can also be used to store data in place of, or in concert with, external databases.

Web Application Server – An industry term for a tool that allows one to serve an application to clients over the Web typically tied to a back-end database. Lasso Professional Server is a Web Application Server which serves Lasso-enabled Web sites. Web Application Server is synonymous with Web Data Engine.

Web Data Engine – LassoSoft's trademark name, synonymous with Web Application Server, but specific to the Lasso product line.

Web Serving Folder – The folder from which a Web site is served (also known as the Web server root). This folder typically contains HTML files, Lasso pages, JPEG or GIF image files, style sheets, and more. Note that if multiple Web sites are served from the same Web server, they might each have their own Web serving folder.

Administration and Security

Administrator – A person who has privileges to alter the setup of Lasso Professional Server. In Lasso Professional Server, there are three types of administrators: a single required site administrator, and multiple optional group administrators.

Anonymous User – Any client which visits a Lasso-based Web site without authenticating, or for which there exists no matching username and password. This client inherits the permissions specified for the AnyUser group.

Group – A collection of users in Lasso Security. Permissions are assigned exclusively to groups. A user can belong to one or more groups, and inherits the permissions from all of the groups to which they belong.

Group Administrator – A user who has the privilege to assign other users to a specified group.

Server Administration – The interface and functionality for configuring Lasso Professional Server. Available as a LassoApp entitled ServerAdmin.LassoApp loaded on-demand from the LassoApps folder in the Lasso Professional Server application folder.

Server Administrator – The person who has access to the server-wide settings including creating new sites. There is only one Site Administrator specified by a single username and password.

Site Administration – The interface and functionality for configuring each site in Lasso Professional Server. Available as a LassoApp entitled SiteAdmin.LassoApp loaded on-demand from the LassoApps folder in the Lasso Professional Server application folder.

Site Administrator – A person who has access to all configuration and security settings for a single site. There is only one Site Administrator per site specified by a single username and password.

Lasso Security – The built-in system for securing data-driven Web sites powered by Lasso Professional Server. Lasso Security allows data sources, databases, tables, records, fields, tags, and Lasso pages to be secured.

Permissions – Permissions are granted to groups and define what tags a user who belongs to the group can execute, and what data sources the user can access. Users inherit the permissions from all of the groups to which they belong.

Privileges – Privileges are granted to administrators and define what administrative and security settings each administrator can access. Administrators are generally granted privileges to modify the permissions of one or more users and groups.

Tag Category – A set of related tags which can be enabled or disabled as a group. Tag categories include Math, Date, String, etc. The names for Lasso tags in a category are often related by sharing the same prefix before an underscore character. (e.g. [File_Control] and [File_Delete] are both members of the File tag category.)

User – A specific visitor of a Web site with a unique username and password. A user is asked for authentication information by the Web browser. A user can belong to one or more groups and inherits the permissions from all of the groups to which they belong.

Internet Technologies

CGI or Common Gateway Interface – Commonly used as a generic term for any server-side technology that processes URLs and HTML form inputs and returns formatted results.

FTP or File Transfer Protocol – A client/server protocol for exchanging files with a host computer.

HTML or HyperText Markup Language – The tag-based markup language for the World Wide Web. HTML can be created using a text editor or a visual authoring tool.

HTTP or HyperText Transfer Protocol – The protocol which governs how a Web browser communicates with a Web server.

HTTPS or HyperText Transfer Protocol, Secure – Version of HTTP developed by Netscape for secure transactions. URLs that begin with “HTTPS” indicate that the SSL protocol is being used.

SMTP or Simple Mail Transfer Protocol – The protocol which governs how email clients communicate with email servers. SMTP support allows Lasso to send email directly to an SMTP email server.

SSL or Secure Socket Layer – A protocol designed by Netscape to enable encrypted, authenticated communications across the Internet.

WAP or Wireless Application Protocol – The protocol which governs how a wireless browser, such as in a cell phone or PDA, communicates with a Web server.

WML or Wireless Markup Language – A tag-based markup language based on XML which allows pages to be formatted for display on a wireless browser.

XML or Extensible Markup Language – A tag-based markup language which is similar to HTML but can be extended with new tags for special purposes. This language is commonly used for data exchange.

XML-RPC or XML Remote Procedure Call – A communication method using HTTP as the transport and XML as the encoding. XML-RPC allows complex data structures to be transmitted, processed, and returned.

Programming

Array – A series of values stored sequentially. The values can be stored and retrieved by index. This allows many related values to be stored and retrieved as a group. For example, the days of the week might be defined as an array, and then the name of the third day of the week retrieved:

```
[Variable: 'Week'=(Array: 'Sunday', 'Monday', 'Tuesday', 'Wednesday',
    'Thursday', 'Friday', 'Saturday')]
```

```
[Array_Get: 'Week', 3] → Tuesday
```

Classic Lasso or Classic Method – The programming method which uses Lasso commands within URLs or HTML forms. These commands are passed from the page in which they are specified to a response page which formats the result of the commands. This method is commonly contrasted with Inline Lasso or Inline Method.

Command Tags – Lasso tags which can be embedded within URLs, HTML forms, or [Inline] tags which instruct Lasso to perform a command or action. Command tags are named starting with a hyphen (e.g. -Database or -KeyField). Their value is specified differently depending on how the command tag is used.

Compound Expression – A form of Lasso syntax that incorporates traditional bracket syntax and LassoScript. For example, [Encode_HTML: {if: (\$Var == Null); Return: 'Default'; Else; Return: \$Var }->Run], uses a LassoScript conditional statement and variables within the [Output] tag. LassoScript within traditional syntax is always contained within {...} characters.

Container Tags – Lasso tags which perform an operation on a portion of a page. For instance, the [Records] ... [/Records] tags will repeat the code contained within them for every database record returned by a Lasso action, but will still allow the code contained within the container tags to be processed.

Data Type – The type of data for any value used by Lasso (e.g. string, integer, etc.). Operations and member tags behave differently depending on the Data Type of the values they are passed. For more information see the *Lasso 8 Tag Language* chapter *Data Types* section in the Lasso 8.5 Language Guide.

Decimal – Any number that contains a decimal place. -300.20, 1.0, 2000.40 are all decimals. Decimal literals should never be surrounded by quotes.

Encoding – The process of converting characters to their hexadecimal equivalent in order to accurately represent them to the client in a Web browser, URL, or XML file. Values returned by Lasso can be returned using a variety of different encoding methods to ensure the proper results.

Expressions – The building blocks of values in Lasso. Expressions include sub-tags, mathematical operators, and member tag calls. Any place that a string or integer can be used as a parameter value for a tag, an expression can be used instead. For more information see the *Lasso 8 Tag Language* chapter *Expressions and Symbols* section in the Lasso 8.5 Language Guide.

Lasso page – A text file that contains Lasso tags which are read for instructions on how to process and format any specific action or request.

Inline Lasso or Inline Method – The programming method accomplished using the [Inline] tag which provides the ability to specify commands and the formatted results of those commands within a single Lasso page. This method is commonly contrasted with Classic Lasso and can provide enhanced security and code protection.

Integer – A whole number that does not contain a decimal place. -3000, 1, 200 are all integers. Integer literals should never be surrounded by quotes.

Keyword – A parameter for an Lasso tag that affects the output of the tag. Keywords always start with a hyphen. For example, the -Encode... keywords in substitution tags are used to specify how the output of the tag is encoded so HTML tags are rendered properly.

LassoScript – A method of specifying Lasso tags within a single container <?LassoScript ... ?>. This allows for a block of programming logic to be formatted in a more readable fashion without any square brackets. Usually used by developers who are familiar with other programming languages such as JavaScript. The following example performs a series of math operations in a LassoScript.

```
<%LassoScript
  Var: 'Total' = 0;
  Var: 'Total' = (Var: 'Total') + (Var: 'Item One Price');
  Var: 'Total' = (Var: 'Total') + (Var: 'Item Two Price');
  Var: 'Total' = (Var: 'Total') + (Var: 'Shipping');
%>
```

Map – A series of values stored and retrieved by name. This allows many values to be stored and retrieved as a group, but still referenced individually by name. A record for a user of your site might be constructed as follows where the first name is retrieved:

```
[Var: 'User'=(Map: 'First'='Bob', 'Last'='Smith', 'Phone'='555-1212')]

[Map_Get: 'User', 'First'] → 'Bob'
```

Member Tags – Tags which are associated with a specific data type. The function of these tags is defined by the data type with which they are associated. A member tag is called as part of an expression by using the member operator `->`. The tag is specified to the right of the operator and the value which the tag is to operate upon is defined to the left of the operator. The following expression gets the third element of the array returned by the variable `myArray`.

```
[Encode_HTML: (Var: 'myArray') -> (Get: 3)]
```

Namespace – A collection of tags with a common purpose. The namespace precedes the tag name with a hyphen. For example, `[String_Replace]` is the `Replace` tag within the `String_` namespace.

Parameter – A component of a tag that follows the tag name and defines the functionality of the tag. Parameters can be simple keywords or complex expressions with many sub-tags. Named parameters and keywords always start with a hyphen.

Pipes – A pipe is a technique for passing information from one process to another. Pipes allow data to be sent to background processes for asynchronous processing, allowing threaded systems to be created for faster performance.

Process Tags – Lasso tags which perform an action, but do not return any result to the client. For instance, the `[Email_Send]` tag sends an email message when the Lasso page that contains it is loaded, but does not show any information regarding the email to the client.

References – A method of passing parameters from one action to another where an alias to a parameter value is passed rather than the actual value. Returning data by reference reduces memory usage and increases code execution speed.

Semaphores – A variable with a value that indicates the status of a common resource. A process needing a resource can check a semaphore to determine the resource's status and then decide how to proceed. Semaphores allow safe use of the same resources across multiple threads.

String – A sequence of letters, numbers, and other characters. Strings with letters are enclosed in single quotes, e.g. `'This is a string'`. Strings with numbers which are to be interpreted as an integer, decimal, or date type should not be enclosed in quotes lest they be interpreted as a non-numeric text string.

Substitution Tags – Lasso tags which take one or more parameters and return a result that is substituted in place of the tag in the page which will be output to the client. For instance: `[Date_GetCurrentDate]` is replaced by the current date and time.

Sub-Tags – Substitution tags used as part of an expression. The sub-tag is specified exactly like a normal substitution tag except that the square brackets which delimit the tag are replaced by parentheses. Sub-tag expressions are usually used as parameters for another tag. In the following example the `[Field]` tags are sub-tags used as parameters for the `[Math_Add]` tag:

```
[Math_Add: (Field: 'Price'), (Field: 'Shipping')]
```

Symbols – Used to perform math or string operations, or logical comparisons in conditional `[If]` tags and expressions (e.g. `=`, `-`, `+`). The behavior of symbols change based on how their parameters are defined and based on the data type of the parameters being modified. For example, `('1'+1')` returns 11 since both param-

eters are of the string data type (in single quotes).

(1 + 1) returns 2 since both parameters are of the integer data type. For more information see the *Lasso 8 Tag Language* chapter *Expressions and Symbols* section in the Lasso 8.5 Language Guide.

Variable – A temporary storage place for a programmer-defined value. Values can be stored in variables and then referenced later in a page. Or, using a session, values can be stored in a variable and then referenced on subsequent pages accessed by the same client. The following example sets a variable and then retrieves it within an HTML tag:

```
[Var: 'URL'='http://www.lassosoft.com']  
  
<a href="[Var:'URL']">LassoSoft</a>
```

Database Technologies

CDML or Claris Dynamic Markup Language – The language of the FileMaker Pro Web Companion. It is basically a subset of an earlier version of Lasso.

Data Source – A database application or database server. Lasso can connect to a data source and access any of the databases which that data source provides.

Database – A collection of tables and their associated records and fields. Every database is hosted by a particular data source.

Field – A container for storing data in a database. In Lasso, field is synonymous with the SQL term column.

JDBC or Java Database Connectivity – JDBC™ technology, produced by Sun Microsystems, is the standard API that allows access to virtually any tabular data source using the Java™ programming language. A JDBC-compliant data source is any data source that has a JDBC driver developed for it.

Key Field – A field whose values (called **Key Values**) uniquely identify particular records. Used to establish relations between different tables. In Lasso, -KeyField is synonymous with -KeyColumn. For FileMaker Pro databases, -KeyField_Value is synonymous with -RecordID_Value.

Record – A unique collection of fields within a database distinguished by a unique key field value. In Lasso, record is synonymous with the SQL term row.

SQL or Structured Query Language – The language which is used to query and build SQL compatible databases. Lasso automatically creates SQL statements based on the criteria specified, or allows one to submit custom SQL queries.

Table – A collection of records that share common fields. Every Lasso action must be performed on a specific table within a database. In Lasso, table is synonymous with the FileMaker Pro term layout.

Extending Lasso

API or Application Programming Interface – A set of shared libraries, documentation, header files, and examples that allow a third party to create a tool which interacts with an existing application.

Lasso Connector Protocol – Defines the rules that govern communication between a Lasso Web server connector and Lasso Service. This protocol is documented in the Lasso 8.5 Language Guide so third parties can create custom Lasso Web server connectors.

LCAPI or Lasso C/C++ API – This API allows third parties to develop Lasso tag modules or Lasso data source connectors using the C/C++ programming language. Documentation is available in the Lasso 8.5 Language Guide.

LassoScript or Lasso Dynamic Markup Language – Custom tags, data types, asynchronous processes, and data sources can be created using Lasso. Documentation is available in the Lasso 8.5 Language Guide.

LJAPI or Lasso Java API – This API allows third parties to develop Lasso tag modules or Lasso data source connectors using the Java programming language, and documentation is available in the Lasso 8.5 Language Guide.

B

Appendix B

License Agreement

PLEASE READ THIS LICENSE CAREFULLY BEFORE INSTALLING OR USING LASSO SOFTWARE. BY DOWNLOADING, INSTALLING, AND/OR USING THE SOFTWARE, YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, YOU ARE NOT AUTHORIZED TO DOWNLOAD, INSTALL, AND/OR USE THIS SOFTWARE.

- 1 License** – The Lasso products, applications, components, modules, application programming interfaces (APIs), and programming language including Lasso Service, Lasso data source connectors, Lasso Web server connectors, Lasso modules, Site Administration interface, and Lasso Dynamic Markup Language (LassoScript), whether on disk, in read only memory, or on any other media or networked storage device (the “Lasso Software”) and related documentation (the “data”) are licensed to you by LassoSoft, LLC (the “Author”). You own the disk on which the Lasso Software and data are recorded but the Author and/or the Author’s Licenser(s) retain title to the Lasso Software and related documentation.

This License allows you to install and use a single copy of the Lasso Software and data on a single computer and make one copy of the Lasso Software and data in machine-readable form for backup purposes only. You must reproduce on such copy the Author’s copyright notice and any other proprietary legends that were on the original copy of the Lasso Software and related documentation. You may transfer all your license rights in the Lasso Software and related documentation, the backup copy of the Lasso Software and related documentation, and copy of this License to another party, provided the other party reads and agrees to accept the terms and conditions of this License and a transfer of ownership letter, listing contact information for both parties, is sent to the Author.

- 2 Upgrading** – In order to facilitate porting existing solutions to the current version of the Lasso Software, this License allows the prior version and the current version of the software to be run concurrently for a maximum of 45 days from the purchase date of the Lasso Software. After 45 days the prior version must be destroyed and no longer used.
- 3 Evaluation Versions** – Evaluation versions are provided for one-time 30-day evaluation and initial product testing use. Evaluation versions, including accompanying documentation, are not licensed for use beyond the 30-day evaluation period.
- 4 Restrictions** – The Lasso Software and related documentation contains copyrighted material, trade secrets and other proprietary material and in order to protect them you may not decompose, reverse engineer, disassemble or otherwise reduce the Lasso Software to a human-perceivable form without the Author’s written permission. Open source components may be modified and made available to others free of charge provided that said modifications are designed to only work with the most current version of the Lasso software. Developers are explicitly forbidden from reverse engineering and/or performing modifications to open source components distributed with the Lasso software in such a way as to make features within said components operate within versions of Lasso software where said features are not designed by LassoSoft to operate.

- 5 Termination** – This License is effective until terminated. You may terminate this License at any time by destroying the Lasso Software and related documentation and all copies thereof. This License will terminate immediately without notice from the Author if you fail to comply with any provision of this License. Subject to the terms of the Upgrading section above, this License will terminate upon upgrading, whether via free or paid upgrade, to a newer version of the Lasso software. Upon termination you must destroy the Lasso Software, and related documentation and all copies thereof.
- 6 Disclaimer of Warranty on Lasso Software** – You expressly acknowledge and agree that use of the Lasso Software and related documentation is at your sole risk. The Lasso Software and related documentation are provided AS IS and without warranty of any kind and the Author EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE AUTHOR DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE LASSO SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE LASSO SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN THE LASSO SOFTWARE WILL BE CORRECTED.
- FURTHERMORE, THE AUTHOR DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE LASSO SOFTWARE AND DATA OR RELATED DOCUMENTATION IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY THE AUTHOR OR A REPRESENTATIVE AUTHORIZED BY THE AUTHOR SHALL CREATE A WARRANTY OR IN ANY WAY INCREASE THE SCOPE OF THIS WARRANTY. SHOULD THE LASSO SOFTWARE PROVE DEFECTIVE, YOU (AND NOT THE AUTHOR OR A REPRESENTATIVE AUTHORIZED BY THE AUTHOR) ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.
- 7 Limitation of Liability** – UNDER NO CIRCUMSTANCES INCLUDING NEGLIGENCE, SHALL THE AUTHOR BE LIABLE FOR ANY INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES THAT RESULT FROM THE USE OR INABILITY TO USE THE LASSO SOFTWARE OR RELATED DOCUMENTATION AND DATA, EVEN IF THE AUTHOR OR AN AUTHORIZED REPRESENTATIVE OF THE AUTHOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU. In no event shall the Author's total liability to you for all damages, losses, and causes of action (whether in contract, tort (including negligence) or otherwise) exceed the amount paid by you for the Lasso Software and data.
- 8 Complete Agreement** – This License constitutes the entire agreement between the parties with respect to the use of the Lasso Software, related documentation and data, and supersedes all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter. No amendment to or modification of this License will be binding unless in writing and signed by the Author or a duly authorized representative of the Author.



Appendix C

Copyright Notices

Open LDAP

The OpenLDAP Public License

Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

- 1 Redistributions in source form must retain copyright statements and notices,
- 2 Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and
- 3 Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS ``AS IS'' AND ANY EXPRESSED 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 OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE 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.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved.

Permission to copy and distribute verbatim copies of this document is granted.

Regular Expressions

PCRE code written by Philip Hazel <ph10@cam.ac.uk>

Copyright © 1997-2000 University of Cambridge

Permission is granted to anyone to use this software for any purpose on any computer system, and to redistribute it freely, subject to the following restrictions:

- 1 This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
- 2 The origin of this software must not be misrepresented, either by explicit claim or by omission.
- 3 Altered versions must be plainly marked as such, and must not be misrepresented as being the original software.
- 4 If PCRE is embedded in any software that is released under the GNU General Purpose Licence (GPL), then the terms of that licence shall supersede any condition above with which it is incompatible.

Compression Algorithm

zlib.h – interface of the zlib general purpose compression library

version 1.1.3, July 9th, 1998

Copyright © 1995-1998 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 Mark Adler

loup@gzip.org madler@alumni.caltech.edu

The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files <ftp://ds.internic.net/rfc/rfc1950.txt> (zlib format), [rfc1951.txt](ftp://ds.internic.net/rfc/rfc1951.txt) (deflate format) and [rfc1952.txt](ftp://ds.internic.net/rfc/rfc1952.txt) (gzip format).

MD5 Message-Digest Algorithm

Copyright © 1990, RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it is identified as the “RSA Data Security, Inc. MD5 Message-Digest Algorithm” in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as “derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm” in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided “as is” without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

iText Java-PDF Library

Support for the PDF file format in Lasso Professional Server is derived from the iText Java-PDF Library by Bruno Lowagie and Paulo Soares, © 1999-2002 by Bruno Lowagie. All Rights Reserved. This library is made available under the Mozilla Public License Version 1.1. A copy of this License is available at <http://www.mozilla.org/MPL/>.

Software distributed under the Mozilla Public License is distributed on an “AS IS” basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

ImageMagick

Copyright © 1999-2003 ImageMagick Studio, a non-profit organization dedicated to making software imaging solutions freely available.

This software and documentation is provided “as is,” and the copyright holders and contributing author(s) 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.

The copyright holders and contributing author(s) will not be held liable for any direct, indirect, special or consequential damages arising out of any use of the software or documentation, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, documentation and executables, for any purpose, without fee, subject to the following restrictions:

- 1 The origin of this source code must not be misrepresented.
- 2 Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
- 3 This Copyright notice may not be removed or altered from any source or altered source distribution.

The copyright holders and contributing author(s) specifically permit, without fee, and encourage the use of this source code as a component for supporting image processing in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

libCURL

Copyright © 1996 - 2001, Daniel Stenberg, <daniel@haxx.se>. All rights reserved.

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, and/or sell copies of the Software, and to

permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies 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 NON-INFRINGEMENT 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 THIS 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 this Software without prior written authorization of the copyright holder.

OpenSSL

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright © 1998-2001 The OpenSSL Project. All rights reserved.

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

- 1 Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- 2 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.
- 3 All advertising materials mentioning features or use of this software must display the following acknowledgment:

“This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit.
(<http://www.openssl.org/>)”

- 4 The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
- 5 Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.
- 6 Redistributions of any form whatsoever must retain the following acknowledgment:

“This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit
(<http://www.openssl.org/>)”

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED 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 OpenSSL PROJECT OR ITS 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.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

SSLeay

Copyright © 1995-1998 Eric Young (eay@cryptsoft.com). All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com). Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- 1 Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
- 2 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.
- 3 All advertising materials mentioning features or use of this software must display the following acknowledgement:

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)." The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related :-).

- 4 If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

"This product includes software written by Tim Hudson (tjh@cryptsoft.com)."

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``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 AUTHOR 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.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

D

Appendix D Index

A

- Administration 28, 110
- Administrator 229, 230
- Administrator Settings 90, 116
- Admin Folder 45, 60, 72
- Aliases 154, 155, 156
- Anonymous User 168, 230
- AnyUser Group 166
- Apache 1.3 34
- Apache 2 34, 35
 - Installation 51
- API 234
- Application Files 44
- Architecture 27, 229
- Array 232
- Asterisk 156
- Authoring Tools 27

B

- Browse
 - Add Options 212
 - List Options 214
 - Search Options 211
 - Update Options 217
- Browsing Databases 209
- BSD 33

C

- Caches 206
- Cache Settings 206
- Cascading Style Sheets 34, 50
- CDML 234
- CGI 231
- Classic Lasso 232
- Classic Lasso Syntax 117
- Client 229
- Client Device Support 26
- Code 14
- Command Tags 232

- Comma Delimited Format 219, 221
- Compiled Applications 28
- Compound Expressions 232
- Connectors 128
- Container Tags 232
- Cookies 34, 50
- Copyright Notices 237
- Critical Errors 107, 201
- Cross References 14
- Customer Support 15
- Custom Tags 92

D

- Databases 234
 - Creating 222
 - Duplicating 223
- Database Browser 208
- Database Builder 208, 222
- Database Permissions 170
- Database Schema 155, 171
- Database Security Operation 169, 171
- Data Source Hosts 139
- Data Sources 126, 234
- Data Source Connectors
 - Updating 128
- Data Source Hosts 130, 136
- Data Source Integration 26
- Data Types 232
- Decimal 232
- Default Site 95
- Definitions 14
- Delete Expired at Startup 122
- Delete Expired Frequency 122
- Details 108, 202
- Developer 229
- Distributed Architecture 58
- Documentation 12, 15
- Documentation Conventions 14
- Documentation Folder 45, 59, 72

E

Email 193
 Email Queue 193
 Email Setup 196
 Encoding 232
 Errors 106, 200
 Error Database 108, 202
 Error Setup 107, 201
 Evaluation 16
 Events 197
 Event Queue 197
 Exporting Data 220
 Exporting Lasso Setup Files 93, 124
 Expressions 232
 Extended Configuration 38, 55
 Extensibility 30

F

Fields 157, 234
 Creating 226
 Field Filters 158
 Field Permissions 173
 Field Security Operation 173
 FileMaker Licensing 129
 FileMaker Pro 128
 Configuration Guidelines 130
 Requirements 128
 FileMaker Server 9
 Configuration Guidelines 133
 FileMaker Server Advanced 132
 Configuration Guidelines 133
 Creating Connections 134
 Requirements 132
 File Extensions 43, 58
 File Paths 14
 File Permissions 175
 File Root 176
 File Security Operation 176
 File Tags 177
 File Uploading 176
 Filters 158
 Format Files 177, 232
 Format File Settings 119
 FTP 231

G

GroupAdmin.LassoApp 168
 Groups 230
 Adding 165
 Updating Database Permissions 170
 Updating File Permissions 176
 Updating Groups 166
 Updating Host Permissions 169
 Updating Table Permissions 172
 Updating Tag Permissions 174
 Group Administration LassoApp 168
 Group Administrator 168, 230

H

Host Permissions 168
 HTML 231
 HTTP 231

httpd.conf 47
 HTTPS 231

I

Import/Export 92, 124
 Importing Data 218
 Importing Lasso Setups 93, 124
 Import File 219
 Import Settings 219
 Include Files 177
 Indices 227
 Creating 227
 Industry Standards 30
 Initialization 35, 52
 Initializing Lasso Professional Server 52
 Inline Lasso 232
 Installation
 Mac OS X 34
 Windows 50
 Installation Contents 44, 59
 Installing Lasso Professional Server 34, 50
 Integer 232
 Introduction 11
 ISAPI Filter 57
 iText 45, 60, 72

J

Javascript 34, 50
 Java Installation 50
 Java Settings 123
 Java Virtual Machine 49
 JDBC 234
 Requirements 136
 Setup 136
 JDBCDrivers Folder 45, 59, 60, 72
 JDBC Data Sources 135
 JRE 49, 50

K

Keyword 232
 Key Field 234

L

Languages and Methodology 28
 Lasso 8 229
 Lasso 8 Reference 13
 LassoApps 202
 Advantages 202
 Building 203
 Characteristics 203
 LassoApp Builder 202
 Preparing Source Files 203
 Using 204
 LassoApp Settings 205
 Lasso Connector for MySQL 138
 LassoErrors.txt 108, 202
 LassoModules Folder 45, 60, 73
 Lasso MySQL Databases 153
 LassoScript 232
 LassoService 46, 73
 LassoService.exe 60
 LassoService.pid 45, 73

- LassoService.sh 45, 73
- LassoStartup Folder 46, 60, 73
- Lasso 8 229
- Lasso 8 Language Guide 12
- Lasso 8 Product Line 11
- Lasso Administration 110, 231
 - Accessing 84, 110, 208
 - Using 86, 111
- Lasso Architecture and Extensibility 19
- Lasso Connector 229
- Lasso Connector for Apache 11, 41
- Lasso Connector for Dreamweaver 12
- Lasso Connector for GoLive 12
- Lasso Connector for IIS 11, 57
- Lasso Connector for WebSTAR V 12
- Lasso Connector Protocol 234
- Lasso Data Source Connector 230
- Lasso Data Source Connectors 127
- Lasso Errors 106, 200
- Lasso Fundamentals 17
- Lasso Internal Databases 87, 113
- Lasso Modules 230
- Lasso Professional 8 11, 17, 229
 - Architecture 18
 - Phases of Operation 20
 - Running 37, 54
- Lasso Professional 8 Setup Guide 12, 13
- Lasso Security 161, 231
- Lasso Service 11, 27, 230
- Lasso Service Console 108, 202
- Lasso Studio 8 12, 229
- Lasso Tag Module 230
- Lasso Tags 14
- Lasso Talk Mailing List 15
- Lasso User 44
- Lasso Web Server Connector 230
- LCAPI 234
- License Agreement 235
- Line Endings 219, 221
- LJAPI 234
- Local System User 59

M

- Mac_OS_X_Tips.pdf 45, 72
- Mac OS X 33
- Mac OS X Installation 33
- Map 233
- Master Site 95
- Maximum Expiration Time 122
- Member Tags 233
- Memory Session Driver 121
- Microsoft Internet Explorer 50
- MySQL
 - Requirements 138
 - Security 138
- MySQL 4.1/5.x Character Sets 140
- MySQL Compatibility 123

O

- Open Source 27
- Operating Systems and Web Servers 26

P

- Parameter 233
- Permissions 231
- Pipes 233
- Ports 37, 54, 68
- Privileges 231
- Process Tags 233
- Product Features 26
- Product Line 229
- Programming 29
- Program Files 59
- Property Inspectors 12

R

- Record-Level Security 178
- Records 234
 - Adding 212
 - Deleting 216
 - Listing 213
 - Searching 210
 - Updating 216
- Record Detail 215
- References 233
- Remote Apache Configuration 42
- Remote IIS Configuration 58
- Remote WebSTAR V Configuration 43
- Resetting the Global Administrator Password 85
- Response Files 177

S

- Scheduling Events 198
- Scripts 171
- Security 28
 - Tips 179
- Semaphores 233
- Sending Email 195
- Sessions 120
 - Memory 121
- Session Tuning
 - Delete Expired at Startup 122
 - Delete Expired Frequency 122
 - Maximum Expiration Time 122
 - MySQL Compatibility 123
- Setup Section
 - Lasso Administration 89, 115, 160
- Sites
 - Add Panel 99
 - Creating 99
 - Criteria 95
 - Default 95
 - Independence 98
 - List Page 99
 - Managing 103
 - Master 95
 - Modules and Folders 97
 - Routing 96
 - Setting Up 95, 98
 - Starting 101
 - Stopping 101
 - Site Administrator 163, 231
 - Site Administrator Password 116
 - SMTP 231

SMTP Server 196
SQL 234
SQL Stop Words 155
SSL 231
StartupItems 46
Start Menu 59
String 233
Sub-Tags 233
Substitution Tags 233
Support Central 15
Symbols 233
Syntax Checking and Conversion 12
System Requirements 33, 49

T

Table Batch Change 131, 155
Tables 156, 234
 Creating 224
 Duplicating 225
Table Permissions 172
Table Security Operation 173
Tab Delimited Format 219, 221
Tags 91
Tag Categories 92, 231
Tag Dependencies 92, 174
Tag Permissions 174
Tag Security Operation 175
Testing Solutions 181
Tools Folder 46, 73
Troubleshooting 35, 52

U

UFS 33
Uninstalling Lasso Professional Server 47, 61
Unix Files 47, 74
Use MySQL 4.1/5.x Character Sets 140
User 229, 231
Users 166
 Adding 167
 Updating 167
User Authentication 180
User Listing 167
User Types 229

V

Variable 234
Visitor 229

W

WAP 231
Warnings 108, 202
Watchdog Script 45, 73
WebSTAR V 35
Web Application Server 230
Web Browser Requirements 34, 50
Web Companion 129
 Setup 129
Web Data Engine 230
Web Server Root 47, 60, 74, 230
Web Server Security 177
Windows Installation 49
Windows System Files 61

WML 231

X

X11, Installing 34
XML 219, 221, 231
XML-RPC 231