

FOUNDATION

# PHP 5 for Flash

*Build Database-Driven Rich Internet Applications with Flash, PHP, and MySQL!*

*Covers PHP 5, MySQL 4.1, and ActionScript 2.0*

David Powers

friendsof   
DESIGNER TO DESIGNER™  
an Apress® company

---

# Foundation PHP 5 for Flash

David Powers



---

# Foundation PHP 5 for Flash

Copyright © 2005 by David Powers

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN (pbk): 1-59059-466-5

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Distributed to the book trade in the United States by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013, and outside the United States by Springer-Verlag GmbH & Co. KG, Tiergartenstr. 17, 69112 Heidelberg, Germany.

In the United States: phone 1-800-SPRINGER, e-mail [orders@springer-ny.com](mailto:orders@springer-ny.com), or visit [www.springer-ny.com](http://www.springer-ny.com).  
Outside the United States: fax +49 6221 345229, e-mail [orders@springer.de](mailto:orders@springer.de), or visit <http://www.springer.de>.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710.  
Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit [www.apress.com](http://www.apress.com).

The information in this book is distributed on an "as is" basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is freely available to readers at [www.friendsofed.com](http://www.friendsofed.com) in the Downloads section.

## Credits

**Lead Editors**      **Production Manager**  
Steve Rycroft      Kari Brooks-Copony  
Chris Mills

**Technical Reviewer**      **Production Editor**  
Sham Bhangal      Katie Stence

**Editorial Board**      **Compositor**  
Steve Anglin, Dan Appleman,      Dina Quan

Ewan Buckingham, Gary Cornell,  
Tony Davis, Jason Gilmore,  
Jonathan Hassell, Chris Mills,

**Proofreader**  
Elizabeth Berry

**Indexer**  
Michael Brinkman

**Associate Publisher**      **Artist**  
Grace Wong      Kinetic Publishing Services, LLC

**Project Manager**      **Cover Designer**  
Beckie Stones      Kurt Krames

**Copy Editor**      **Manufacturing Manager**  
Nicole LeClerc      Tom Debolski

---

# CONTENTS

---

<b>About the Author</b> .....	ix
<b>About the Technical Reviewer</b> .....	x
<b>Acknowledgments</b> .....	xi
<b>Introduction</b> .....	xiii
<b>Chapter 1: Getting Ready to Work with PHP</b> .....	1
Accessing external data with Flash .....	2
Choosing the right technology .....	2
What PHP, Apache, and MySQL have to offer .....	4
How everything fits together .....	5
PHP and ActionScript: Distant cousins .....	7
Installing the necessary software .....	9
Setting up on Windows .....	9
Setting up on Mac OS X .....	27
Setting up your work environment .....	38
Getting a first taste of the power of PHP .....	43
<b>Chapter 2: Flash Breaks Free</b> .....	45
Communicating with external sources .....	46
Taking first steps in PHP .....	51
How PHP fits into web design .....	51
The basic grammar of PHP .....	52
Sending feedback from Flash by email .....	64
Progress so far .....	82
<b>Chapter 3: Calculations and Decisions</b> .....	85
Performing calculations with PHP .....	86
Working with arithmetic operators .....	87
Useful math functions .....	91
Performing calculations in the right order .....	93
Combining calculations and assignment .....	95
Making decisions with PHP .....	96
Using if... else conditional statements .....	96
Using comparison operators .....	97

Testing more than one condition . . . . .	99
Using switch for long decision chains . . . . .	102
Using the conditional operator . . . . .	103
Flash application: A multiconverter . . . . .	104
Planning the conversion script . . . . .	105
Building the Flash interface . . . . .	110
Summing up . . . . .	127
<b>Chapter 4: Of Strings and Things . . . . .</b>	<b>129</b>
Manipulating strings with PHP . . . . .	130
How PHP outputs strings . . . . .	130
Changing case . . . . .	135
Working with substrings . . . . .	138
Modularizing code with functions . . . . .	144
Understanding where PHP functions run . . . . .	145
Why roll your own? . . . . .	146
Understanding how PHP and ActionScript functions handle variables . . . . .	147
Returning a value from a function . . . . .	150
Deciding where to put functions . . . . .	151
Completing the multiconverter script . . . . .	151
Formatting the main measurement units . . . . .	152
Handling gallons, pints, and liters . . . . .	155
Dealing with kilograms, pounds, and stones . . . . .	160
Handling meters to feet and yards . . . . .	161
Reviewing the multiconverter project . . . . .	162
Taking the project further . . . . .	163
Dealing with user input . . . . .	164
Trimming leading and trailing whitespace . . . . .	164
Stripping HTML tags . . . . .	165
Removing backslashes . . . . .	165
Using regular expressions to identify patterns . . . . .	166
Fine-tuning the feedback application . . . . .	171
A pause for breath . . . . .	174
<b>Chapter 5: Working Smarter with Arrays and Loops . . . . .</b>	<b>177</b>
Understanding the basics of arrays and loops . . . . .	178
Organizing items in arrays . . . . .	178
Grouping similar items in multidimensional arrays . . . . .	179
Using loops for repetitive tasks . . . . .	180
Creating arrays in PHP . . . . .	180
Indexed arrays: Organizing by number . . . . .	181
Associative arrays: Organizing by name . . . . .	184
Array length: Key to understanding the difference between PHP and ActionScript arrays . . . . .	188
Multidimensional arrays: Nesting arrays . . . . .	189
Zipping through repetitive tasks with loops . . . . .	191
Looping through arrays with foreach . . . . .	191

Using the versatile for loop . . . . .	192
Simple loops with while and do . . . . .	194
Breaking out of loops . . . . .	195
Nesting loops . . . . .	196
Passing information securely with \$_POST . . . . .	198
Why register_globals is so important . . . . .	199
Manipulating arrays . . . . .	201
Slicing and splicing arrays . . . . .	202
Sorting arrays . . . . .	206
Building an RSS feed aggregator . . . . .	209
What an RSS feed does . . . . .	209
Taking a look inside an RSS feed . . . . .	210
Parsing an RSS feed with MagpieRSS . . . . .	212
Displaying the merged RSS feed in Flash . . . . .	223
Eliminating HTML entities that Flash cannot handle . . . . .	229
Deploying the feed aggregator on the Internet . . . . .	230
Progress report . . . . .	231
<b>Chapter 6: PHP and Databases: Packing Real Power Behind Your Applications . . . . .</b>	<b>233</b>
Why MySQL? . . . . .	234
MySQL's shortcomings . . . . .	234
MySQL's strengths . . . . .	236
Choosing the right version of MySQL . . . . .	237
Choosing the right license and cost . . . . .	239
Considering SQLite as an alternative . . . . .	239
SQLite's strengths . . . . .	239
SQLite's shortcomings . . . . .	240
Choosing the right database system . . . . .	240
Installing MySQL on Windows . . . . .	241
Changing the default table type on Windows Essentials . . . . .	253
Starting and stopping MySQL manually on Windows . . . . .	255
Launching MySQL Monitor on Windows . . . . .	257
Configuring MySQL on Mac OS X . . . . .	259
Working with MySQL Monitor (Windows and Mac) . . . . .	261
Creating your first database in MySQL . . . . .	263
Loading data from an external file . . . . .	272
Using MySQL with a graphical interface . . . . .	275
phpMyAdmin: A golden oldie . . . . .	275
MySQL Administrator and MySQL Query Browser: Smart new kids on the block . . . . .	280
Looking ahead . . . . .	282
<b>Chapter 7: Playing with Words . . . . .</b>	<b>285</b>
Building the game's graphical elements . . . . .	286
Using PHP to communicate with the database . . . . .	291
A touch of class to emulate mysqli on all setups . . . . .	292



---

Using the mysqli object-oriented interface . . . . .	292
Building and using PHP 5 classes . . . . .	308
Naming and declaring classes . . . . .	309
Creating class properties . . . . .	309
Using the constructor function . . . . .	310
Setting class methods . . . . .	312
Accessing public methods . . . . .	313
Refining the word selection with SQL and PHP . . . . .	313
Building a scoring mechanism that remembers . . . . .	322
Introducing the Flash SharedObject . . . . .	323
Other ways to enhance the game . . . . .	326
Handling database failures . . . . .	327
Setting different skill levels . . . . .	330
SQLite: An alternative database system . . . . .	331
SQLite basics . . . . .	331
Making sure SQLite has the right permissions . . . . .	333
Rewriting the Hangman PHP script for SQLite . . . . .	341
Comparing MySQL and SQLite . . . . .	343
Ever onward and upward . . . . .	343
<b>Chapter 8: Creating a User Registration Database . . . . .</b>	<b>345</b>
Understanding database types . . . . .	346
Keeping things simple with flat-file databases . . . . .	346
Gaining greater flexibility with relational databases . . . . .	348
Understanding MySQL storage formats . . . . .	353
Choosing the right column type . . . . .	354
Column types in MySQL . . . . .	354
Default values and NULL . . . . .	358
Choosing the right language settings . . . . .	358
Building a user registration system . . . . .	359
Registering users with MySQL . . . . .	359
What if it doesn't work? . . . . .	384
Time to take stock . . . . .	385
<b>Chapter 9: Protecting Your Data with Sessions . . . . .</b>	<b>387</b>
Keeping track with PHP sessions . . . . .	388
The Web is a stateless environment . . . . .	388
How sessions work . . . . .	390
PHP session basics . . . . .	392
Using sessions to restrict access . . . . .	393
Other uses for sessions . . . . .	405
Summary . . . . .	405

---

<b>Chapter 10: Keeping Control with a Content Management System</b> . . . . .	407
The four essential SQL commands . . . . .	408
SELECT . . . . .	408
INSERT . . . . .	410
UPDATE . . . . .	410
DELETE . . . . .	411
Building a simple content management system . . . . .	411
Building the content management interface . . . . .	412
Scripting the application . . . . .	419
Securing the content management system . . . . .	466
Adding an extra column to a table . . . . .	468
A solid foundation has been laid . . . . .	470
<b>Chapter 11: Working with Dates</b> . . . . .	473
How ActionScript, PHP, and MySQL handle dates . . . . .	474
Navigating the minefield of incompatible timestamps . . . . .	475
Creating a timestamp . . . . .	477
Formatting dates in PHP . . . . .	483
Working with dates in MySQL . . . . .	486
Using dates in calculations . . . . .	486
Finding and creating records based on temporal criteria . . . . .	493
Handling dates in user input . . . . .	494
Formatting dates from text input . . . . .	495
Checking a date's validity with PHP . . . . .	495
Building a Flash date selector for MySQL . . . . .	498
Nearly there . . . . .	510
<b>Chapter 12: Working with Multiple Tables and XML</b> . . . . .	513
Designing the table structure . . . . .	514
Deciding the basic requirements . . . . .	514
Normalizing the tables . . . . .	515
Preparing to build the bookstore database . . . . .	518
Getting an overview of the project . . . . .	521
Completing the database structure . . . . .	523
Creating the content management system . . . . .	529
Deciding the basic structure . . . . .	529
Activating the forms with PHP . . . . .	545
Retrieving data from more than one table . . . . .	569
Avoiding ambiguous column references . . . . .	569
Using a full join . . . . .	569
Using a left join to find an incomplete match . . . . .	572
Completing the content management system . . . . .	574
Managing existing book records . . . . .	574
Deleting records from more than one table . . . . .	585
Maintaining referential integrity on deletion . . . . .	587
Updating multiple records . . . . .	594



- Using SimpleXML to parse an XML feed . . . . . 596
- Securing your CMS . . . . . 601
- Displaying the database contents in Flash . . . . . 601
  - Getting the database ready . . . . . 601
  - Communicating with the database through PHP . . . . . 602
  - Building the Flash interface . . . . . 608
  - Creating the ActionScript to load results from the database . . . . . 609
- A long road traveled . . . . . 616
  
- Appendix A: When Things Go Wrong with PHP and MySQL . . . . . 619**
  
- Appendix B: Converting Applications to ActionScript 1.0 . . . . . 641**
  
- Appendix C: Installing Older Versions of MySQL on Windows . . . . . 647**
  
- Appendix D: Using Languages Other Than English in MySQL . . . . . 657**
  
- Appendix E: Essential MySQL Maintenance . . . . . 667**
  
- Index . . . . . 684**

---

## ABOUT THE AUTHOR

---



**David Powers** is a professional writer who has been involved in electronic media for more than 30 years, first with BBC radio and television, and more recently with the Internet. A mild interest in computing was transformed almost overnight into a passion, when he was posted to Japan in 1987 as a BBC correspondent in Tokyo. With no corporate IT department just down the corridor, he was forced to learn how to fix everything himself. When not tinkering with the innards of his computer, he was reporting for BBC radio and television on the rise and collapse of the Japanese bubble economy.

It was back in the UK as Editor, BBC Japanese TV, that David started working with web design. He persuaded the IT department to let him have free run of a tiny corner of the BBC's Internet server; and he built and maintained an 80-page Japanese and English website—first, coding by hand, and then trying all variety of HTML editors, good and bad. He decided to set up his own independent company, Japan Interface (<http://japan-interface.co.uk>) in 1999, and he is actively involved in the development of an online bilingual database of economic and political analysis for Japanese clients of an international consultancy.

This is David's third book for friends of ED/Apress. He co-authored *Foundation Dreamweaver MX 2004* (friends of ED, ISBN: 1-59059-308-1) and *PHP Web Development with Dreamweaver MX 2004* (Apress, ISBN: 1-59059-350-2). David was also the technical reviewer for the highly successful second edition of *Cascading Style Sheets: Separating Content from Presentation* (friends of ED, ISBN: 1-59059-231-X) and *Web Designer's Reference* (friends of ED, ISBN: 1-59059-430-4). He has also translated several plays from Japanese, most recently *Southern Cross*, the final part of a war trilogy by Keita Asari.

---

## ABOUT THE TECHNICAL REVIEWER

---



**Sham Bhangal** has written for friends of ED on new media since the imprint's inception over five years ago. In that time, he has been involved in the writing, production, and specification of just under 20 books.

Sham has considerable working experience with Macromedia and Adobe products, with a focus on web design and motion graphics. Creating books that teach other people about his favorite subjects is probably the best job he has had (ignoring the long hours, aggressive deadlines, lost manuscripts, and occasional wiped hard drives). If he was doing something else, he'd probably be losing sleep thinking about writing anyway.

Sham currently lives in the north of England with his longtime partner, Karen.

---

## ACKNOWLEDGMENTS

---

Writing books is a lonely business, pounding away at the keyboard night and day for months on end, but it's not something you can achieve all on your own. Every member of the editorial and production team deserves thanks for guiding, encouraging, and cajoling me from the initial concept through to completion: Steve Rycroft for planting the original idea for this book in my brain and seeing it through the early stages; Chris Mills and Beckie Stones for putting up with my sometimes interminable phone calls when I needed help in sorting out the forest from the trees; Nicole LeClerc for her sensitive and perceptive copy editing; plus, of course, all the others.

Special thanks must go to Sham Bhangal for the superb job he has done as technical reviewer, subjecting both the text and the code to scrutiny, not only for accuracy, but also for ease of use. On many occasions, he suggested that something buried deep in a chapter would make more sense if brought to the front—and he was right. I originally learned ActionScript from Sham's books, so it was a particular honor to have him work on my first Flash-related book.

I'm also grateful to Al Sparber of Project Seven ([www.projectseven.com](http://www.projectseven.com)) for allowing me to adapt his Uberlink CSS concept for the navigation menu in Chapter 12.

The unsung heroes to whom we should all be grateful are the development teams behind PHP, MySQL, and Flash. Without them, the Web would be a much duller place and there would have been no book to write.

The biggest thanks of all must go to you, the reader, who makes it all worthwhile. If you have just bought this book, I hope it lives up to your expectations. If you have borrowed this book, I hope you enjoy it enough to want to go out and buy a copy of your own.

---

# INTRODUCTION

---

Back in 2001, friends of ED published *Foundation PHP for Flash*. It was very popular, but it has long been out of print, and is now very much out of date. It was written in the days of Flash 5, PHP 4.0, and MySQL 3.23. Since then, a lot of water has passed under the bridge. The release of ActionScript 2.0 in September 2003 and of PHP 5 ten months later represents a significant milestone in the evolution of both languages. Both now have a formal object-oriented programming (OOP) syntax, turning them from being merely useful tools into essential assets for developing rich, interactive web applications. MySQL, the world's most popular open source relational database, has also been making rapid strides. It has gone through two major upgrades (the current version is MySQL 4.1), and work is moving apace on the development of MySQL 5.0, which will add more of the advanced features currently associated with much more expensive commercial systems.

Consequently, *Foundation PHP 5 for Flash* is not a new edition of an old book. It is completely new from the first page to the last. Although it touches only briefly on OOP, the emphasis is on laying a solid foundation of good coding practice, so that when you move on to a more advanced level, you won't have to unlearn any bad habits.

## Using the latest standards, but remaining version neutral

Software developers often move at a rapider pace than administrators with the responsibility for deploying software. Also, not everyone can afford to upgrade every time a new version comes out. So, in writing this book, I have taken particular care to use the latest standards, but to keep them version neutral wherever possible. Where something works only in PHP 5 or MySQL 4.1, I say so clearly and suggest alternative ways of achieving the same result. A special feature of this book is the set of Database classes in Chapter 7 that enable you to use exactly the same code to communicate with a database, whether your hosting company is using the latest versions of PHP and MySQL or it's still stuck in the past with PHP 4 and MySQL 3.23. All the ActionScript is written using ActionScript 2.0, but Appendix B gives advice on how to adapt it to work in Flash MX, and the download files (available from [www.friendsofed.com](http://www.friendsofed.com)) include special versions rewritten in ActionScript 1.0 and saved in Flash MX format.

## Who this book is for

Although this book is part of the Foundation series, it's not aimed at beginners taking their first steps with Flash. It's a book about integrating PHP and MySQL with Flash, so you need to be comfortable moving around the Flash authoring environment. Ideally, you'll also have some experience of ActionScript. You don't need to be able to recite the contents of the ActionScript Dictionary in your sleep, but your progress will be a lot swifter if you have a reasonable grasp of the main

concepts, such as working with loops, arrays, and so on. If you have worked through the ActionScript chapters in *Foundation Flash MX 2004* by Kristian Besley and Sham Bhargal (friends of ED, ISBN: 1-59059-303-0), you should have no difficulty.

Even if you don't have much experience of ActionScript, don't worry. Everything is explained in detail. Rather than just throw a chunk of code at you and tell you to get on with it, my approach is to get you to understand what's happening and why. The other main feature of this book is the way it highlights the differences and similarities between PHP and ActionScript. Both languages use the same basic concepts and structures, but there are often subtle differences that catch out the unwary. So, even if your knowledge of arrays is shaky, by the time you have finished Chapter 5, you should have a firm grasp of how they're used in both PHP and ActionScript.

If you're looking for a half-dozen lines of code that will solve the problems of the universe, then this isn't the book for you. PHP is not a point-and-click language. You need to roll up your sleeves and code by hand. Similarly, the only way to get the best out of a database—not just MySQL—is to learn the basics of database design and Structured Query Language (SQL), the language used to interact with databases. But don't let that put you off. Neither PHP nor MySQL is particularly difficult to learn—that's one reason they're so popular. Creating the code yourself puts great power and flexibility at your fingertips. Instead of being shoehorned into a rigid and, perhaps, inappropriate solution, you gain the freedom to create your own solutions.

If you're still using Flash MX, and you recoil in horror at the thought of Expert Mode in the Actions panel, you're in for a bit of a shock. The dialog boxes in Normal Mode were removed in MX 2004, and it became Expert Mode or nothing. Again, don't worry. By the time you've worked through a couple of chapters, you'll realize that typing in the code directly is much faster. You'll also notice that I don't sprinkle ActionScript all over the place; most of the time it goes on frame 1 of a layer reserved solely for ActionScript. In the projects toward the end of the book, this produces long scripts that may look off-putting at first glance. Take a closer look and you'll quickly realize that they're made up of short code blocks. Keeping them in one place makes updating and maintenance far, far easier than playing "hunt the script" for an action buried in a movie clip nested three levels deep.

What if you're someone with a fair knowledge of PHP, but you're looking for ways to integrate it with Flash? Although I don't set out to teach ActionScript, your knowledge of programming combined with reference to the ActionScript Dictionary in Flash (Help ► ActionScript Dictionary) should make the transition relatively painless. Welcome on board.

## What you need

First of all you need a copy of Flash. Ideally, you should be using MX 2004 or later, but the minimum requirement is Flash MX (see Appendix B for details on how to convert ActionScript 2.0 to work with MX). You also need the Apache web server, PHP, MySQL, and a copy of a graphical interface for MySQL called phpMyAdmin—all available for download free of charge over the Internet. If you're using Mac OS X, Apache and PHP should already be installed on your computer, although you may need to enable them. Installation isn't difficult, but the correct configuration seems to cause problems for many beginners, so I've given detailed instructions for both Windows and Mac in Chapters 1 and 6, highlighting points that seem to trip up many people. The size of downloads varies with your operating system, but some are as big as 20MB, so give yourself plenty of time if you're on a dialup connection.



The only other software needed is a text editor or, preferably, a dedicated script editor. PHP scripts need to be saved as plain text files (but with a `.php` filename extension), so Notepad or TextEdit are perfectly adequate. Your life will be made a lot easier, though, by a script editor that displays line numbers and uses syntax coloring. I find working in Code view of Dreamweaver MX 2004 the most convenient, but other script editors you might wish to consider are TextPad or SciTEFlash (Windows), or BBEdit (Mac). For more information, including where to obtain these programs, see the section “Using a script editor to reduce syntax errors” in Appendix A.

## What you’ll get out of this book

First of all, I hope you get a lot of fun. Although the book is project driven, I’ve structured each chapter around a particular aspect of PHP or working with a database, such as string manipulation, arrays, or joining tables. Hands-on projects are a good way to learn new skills, but it can be a nightmare trying to dig out that vital nugget of information if it’s buried deep in a series of step-by-step instructions. So, wherever possible, I’ve split each chapter into clearly identifiable sections so that you can come back much later and use it as a reference book. The projects are intended to be challenging, rather than the lightweight examples you find in a lot of beginners’ books. By the end of Chapter 12, you will have built the following applications in Flash and PHP:

- An online feedback form
- A tool that converts 24 different types of weights and measures
- An RSS feed aggregator
- A word game with a vocabulary of more than 21,000 words
- A scoring system that remembers a player’s score on returning to a site
- An online registration system
- An intelligent Flash component to format and display dates (intelligent enough to know how many days there are in each month, even in a leap year)
- A multitable relational database

Along the way, you’ll also learn how to protect sensitive parts of your website using PHP sessions, as well as cover the basics of database management, and you’ll take some first steps in OOP with PHP. The focus throughout is on learning how to work with PHP and MySQL, so I’ve deliberately kept the graphical aspects of design simple, but—I hope—elegant.

## Mac-friendly, too

I have a confession to make. For many years, I worked with a Mac enthusiast who loathed Windows so much that he rarely began a conversation without cursing Bill Gates and all his works. (I know some Windows users like that, too, but that’s beside the point.) As a result, he turned me into a Mac-hater (sorry). Then, one day, I read lots of good things about Mac OS X and decided to try it out. I liked what I saw, but it’s difficult to change the habits of two decades of working with DOS and Windows. Still, I’ve tested everything on a PowerBook G4 running OS X 10.3, and I’ve given separate instructions for the Mac wherever appropriate. Fortunately, PHP and MySQL are almost entirely operating system–neutral, so if I don’t give specific instructions for the Mac, it’s not because I have forgotten or am ignoring you.

*Support for PHP 5 on versions of Mac OS X prior to 10.3 appears to be very thin on the ground. If you are running Jaguar or earlier, you will probably be restricted to using PHP 4. Most of the code in this book will run on PHP 4, but the only Mac testing has been done on Panther.*

## Layout conventions used in this book

I've tried to keep this book as clear and easy to follow as possible, so I've used the following text conventions throughout.

When you first come across an important word, it will be in **bold** type, then in normal type thereafter.

I've used a fixed-width font for code, file names, and any other text you need to type in for yourself.

Sections of code that need to be added to an existing script or that deserve special attention are additionally highlighted **like this**.

You'll see menu commands written in the form Menu ► Submenu ► Submenu.

When there's some information I think is really important, I'll highlight it like this:

*This is very important stuff—don't skip it!*

When I ask you to enter code that spills onto two lines **without** using a carriage return, I'll use a code continuation character, like this: ➤

## Getting help when you're stuck

We all make mistakes (although I hope there aren't any in this book!), so it's useful to know where to go for help when things don't turn out as expected. Throughout the text, I've identified the most common pitfalls, but if the answer isn't there, the first place to look for it is this book's page on the friends of ED website ([www.friendsofed.com/books/1590594665/index.html](http://www.friendsofed.com/books/1590594665/index.html)). Check to see if any errata or updates have been posted. The download files are also provided to help you troubleshoot problems. If the download file works, but not yours, it's a clear sign that there's a mistake in your code. Finding it is just a matter of searching patiently.

If you still draw a blank, go through the troubleshooting steps outlined in Appendix A, and make sure that your system is using versions no earlier than the ones used by me and the friends of ED technical team during testing. All the code in the book has been tested on Windows 2000, Windows XP Pro, and Mac OS X Panther, using the following versions:

- **PHP:** 4.3.4, and all versions from 5.0.0 through 5.0.3
- **MySQL:** 3.23.38, 4.0.20, 4.0.21, and all versions from 4.1.5 through 4.1.9
- **Apache:** 1.3.33, 2.0.40, and 2.0.46
- **phpMyAdmin:** 2.5.5 and 2.6.0

*Even if you have the most up-to-date versions on your local computer, your applications could fail when uploaded to a remote server that uses an older configuration. The PHP Database classes in Chapter 7 are designed to get around these incompatibilities, by enabling the same code to work regardless of server configuration, so make sure you upload the correct one.*

If the answer still eludes you, then post a question in the friends of ED support forums at [www.friendsofed.com/forums](http://www.friendsofed.com/forums). Try to give a brief description of the problem; indicate which version of Flash, PHP, and MySQL you are using; and note any remedies that you have tried, but failed. Someone, maybe even me, should soon be along to share the misery—and hopefully provide an answer that puts a smile back on your face.

## Stop the presses: Changes to MySQL 4.1.9

The development team at MySQL, never slouches at any time, were particularly hyperactive during the last few months of this book being written. I managed to incorporate all the relevant changes into the main text, but two more were announced in January 2005, after Chapters 1 and 6 had already been typeset.

### New filename for the Windows Essentials installer

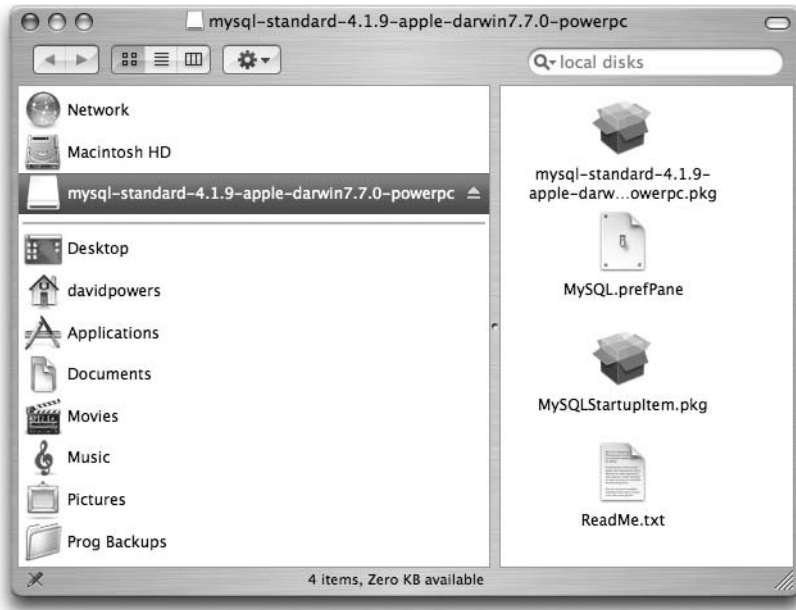
The name of the installation file for MySQL Windows Essentials has been changed. Instead of `mysql-4.1.x-essential-win.msi`, it is now `mysql-essential-4.1.x-win32.msi`. (The value of *x* will depend on the number of the latest release.) The installation instructions in Chapter 6 are otherwise unaffected.



### New MySQL startup method for Mac OS X 10.3

The changes affect steps 3 and 4 of the instructions in the section titled “Installing MySQL on Mac OS X” in Chapter 1, but they do *not* apply to Mac OS X 10.2 (Jaguar).

When the DMG file is mounted on your desktop, you should see something similar to the image shown here. It includes an extra icon labeled `MySQL.prefPane`. As of late January 2005, the `ReadMe.txt` file had not been updated to reflect the changes, so it’s not clear whether `MySQLStartupItem.pkg` will continue to be included in later versions, as it no longer seems to be necessary.



Double-click the icon of the PKG file that begins with mysql-standard-4.1.x and follow the onscreen instructions to install MySQL. When installation is complete, drag the MySQL.prefPane icon onto System Preferences, either in your Dock or in Finder ► Applications. This will open a dialog box asking if you want to install it (see image below). Select whether you want to install it just for yourself or for all users, and then click Install.



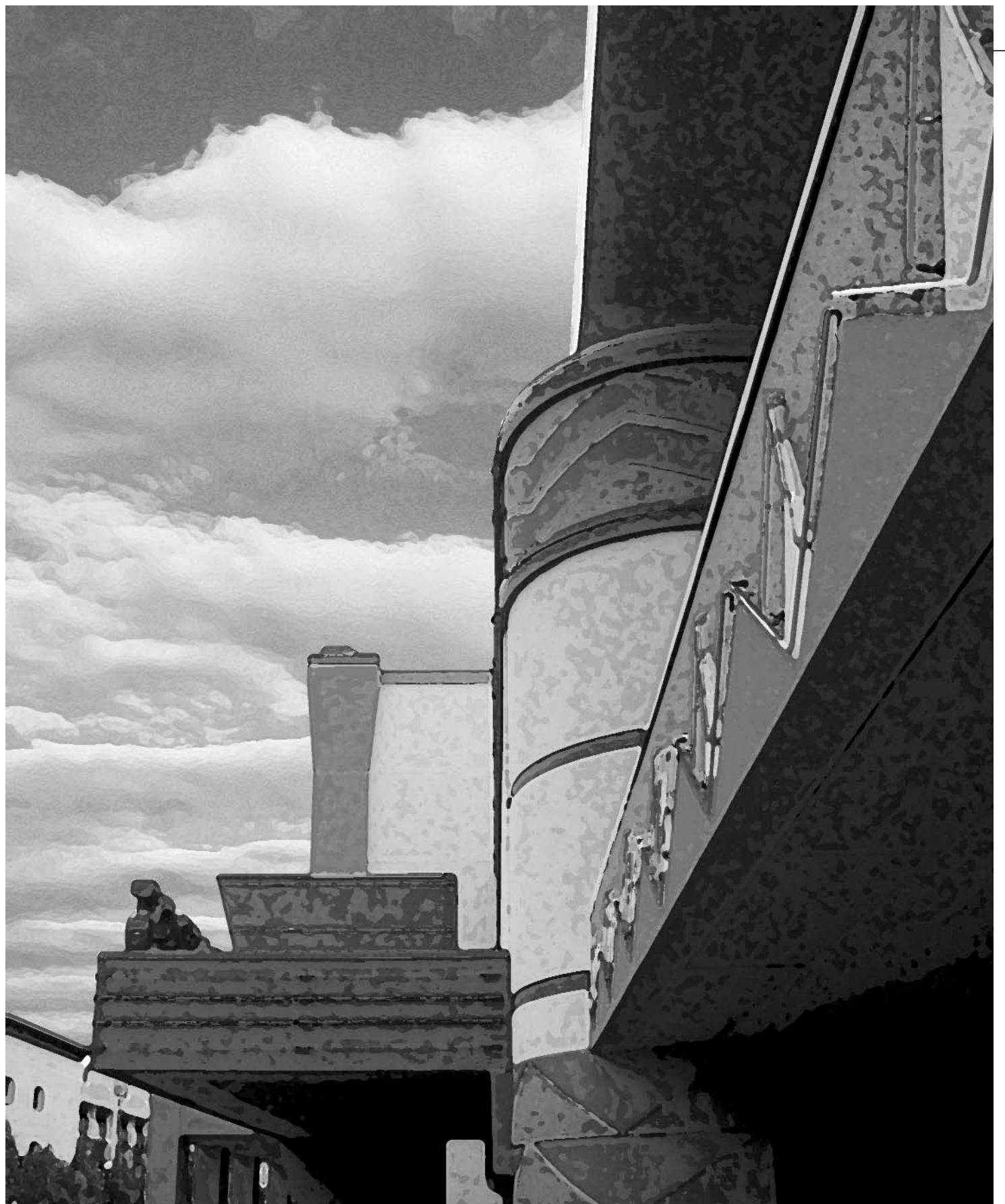
When the MySQL preference pane has been installed, it will open and should show you that MySQL server is running. The preference pane (as shown in the following image) is self-explanatory. It not only provides a convenient Mac-friendly way of starting and stopping MySQL, but also gives you the option to start MySQL automatically whenever your computer starts up.



Close the preference pane in the normal manner by clicking the close button at the top left of the pane. To access it again, open System Preferences. The MySQL icon will be at the bottom of the System Preferences window in the Other section.

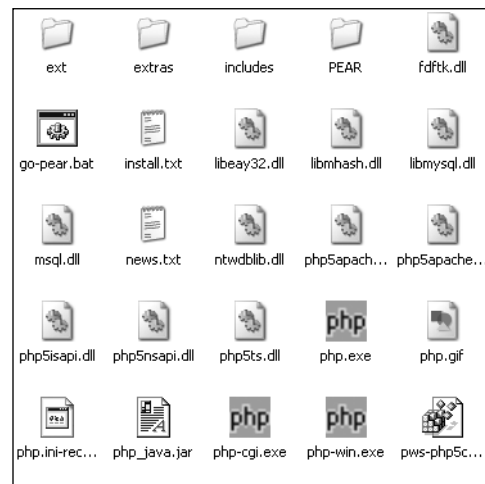
Continue with the installation instructions from step 5.

*If you have installed the MySQL preference pane, always use this method to start and stop MySQL instead of the instructions in the main text of the book. This applies to Mac OS X only. The Windows instructions are not affected.*





## Chapter 1



# GETTING READY TO WORK WITH PHP

### What this chapter covers:

- Expanding Flash's horizons by linking to external data sources
- Choosing the appropriate server-side technology
- Examining similarities and differences between PHP and ActionScript
- Installing Apache and PHP
- Testing the installation
- Setting up your work environment

Flash is fun. It's also extremely powerful. Its power lies not only in its incredible animation capabilities, but also in ActionScript. With the release of Flash MX 2004, ActionScript was upgraded to version 2.0 and became a fully object-oriented programming (OOP) language with support for classes, inheritance, interfaces, and other common OOP concepts. Whether you're making your first personal website, creating immersive online games, or building a serious business site, Flash offers a rich development environment.

Yet, in spite of its power, Flash *on its own* has a severe limitation: the need to embed virtually all content and processing logic within the same medium as the user interface. Even the most minor change means republishing and uploading a new version of the Flash movie. Sensitive content is easily extracted by anyone with a decompiler, several of which are freely available. Most important of all, even with the rapid spread of broadband, download times restrict the optimum size of a movie and therefore the amount of information it can contain. Fortunately, all these limitations are easily overcome by linking your SWF movie to an external data source such as a database or RSS feed. That's what this book is all about, using the latest version of the most popular open source server-side technology, PHP. By the end of this chapter, you should understand how PHP interacts with ActionScript, and you will have installed PHP and the Apache web server on your development computer. (By the way, if you're not sure what an RSS feed is, it's one of those online news feeds that seem to be popping up all over the place these days. Take a quick peek at "What an RSS feed does" in Chapter 5, where you'll be working with two of them.)

## Accessing external data with Flash

A Flash movie—the SWF file—is frequently its own self-contained world. Everything is there within the one file: graphics, timeline animations, and ActionScript functions to control the movie and respond to events initiated by the user. The more you have going on within the movie, the bigger the file gets. To get around this problem, you can load other SWF or JPG files into the main movie only if and when required. Although this can be regarded as accessing external data, it's extremely limiting. Everything has to be fixed in advance and scripted inside the main movie. Real flexibility comes only once you connect to an external database, news feed, or similar source of independent data. This makes possible such things as online reservation systems, news sites, or online forums, all built in Flash. And it's not just a one-way process. Once you've tied up with a server-side technology, you can capture user input, inserting it into a database or sending it across the Internet through email. By tying up Flash with PHP, you open up a whole new range of possibilities and rich Internet applications—not just fixed sources of information or entertainment, but ones that offer real interactivity with the user.

## Choosing the right technology

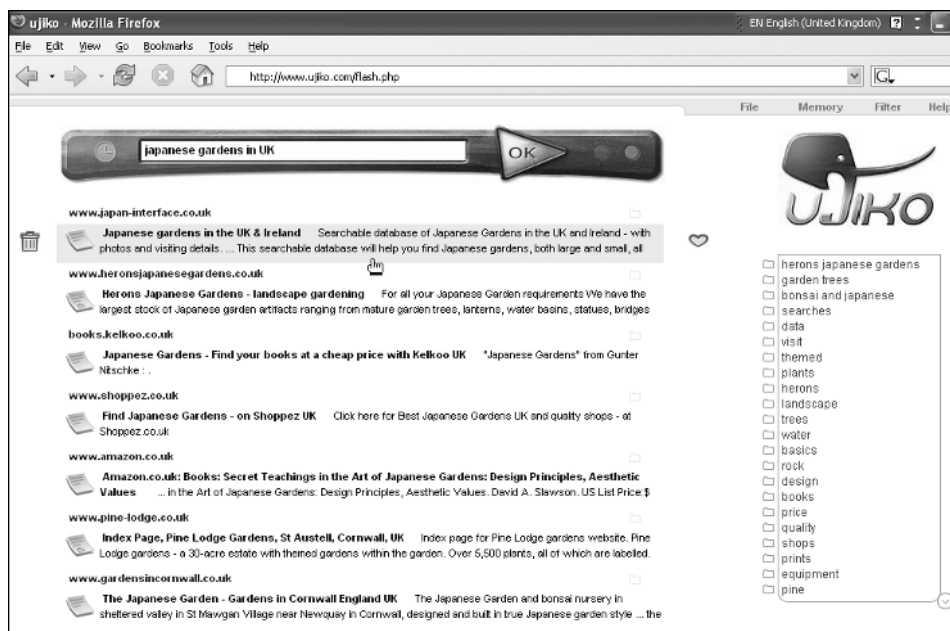
There are three ways of communicating between a Flash movie and an external data source:

- Flash Remoting
- Web services
- Direct interaction with a server-side technology, such as ASP, ASP.NET, ColdFusion, or PHP

Flash Remoting is a proprietary technology developed by Macromedia, the makers of Flash. Its major advantages are speed and the ability of Flash to handle external data as native ActionScript objects. Its major disadvantage is that it requires the installation of a Flash Remoting Server at a cost of approximately \$1,000 per CPU.

Flash MX 2004 Professional introduced the concept of connecting to web services using Simple Object Access Protocol (SOAP) or XML through specialized data components. This led many developers to believe the curtain was coming down on Flash Remoting. However, the Flash web services approach is often slower than Flash Remoting, and it's far from clear whether these technologies will continue to exist side by side, or whether one will prevail. For an in-depth analysis of the advantages and disadvantages of the two methods, see [www.macromedia.com/devnet/mx/flash/articles/ria\\_dataservices.html](http://www.macromedia.com/devnet/mx/flash/articles/ria_dataservices.html).

The third method—direct interaction with Flash using a server-side technology—should not be regarded as “second best” or inferior in any way. The Ujiko search engine at [www.ujiko.com/flash.php](http://www.ujiko.com/flash.php) is built entirely in Flash, and it displayed the results shown in Figure 1-1 just as quickly on a broadband connection as the same search on Google or Yahoo did using ordinary HTML pages. Flash is used not simply to give the search engine a “pretty face,” but also to add extra functionality through using the Flash SharedObject to enable you to refine your searches and set individual preferences. As you hover your mouse pointer over each result, a trash can and a heart appear on either side. Choosing the trash can or the heart removes the result from the list or adds it to your favorites, respectively, helping to refine further searches. A context-sensitive list on the right side of the page presents further keywords to help burrow further down and filter the results until you find what you want. Creating such a rich user experience with a standard HTML interface would be very difficult, if not impossible.



**Figure 1-1.** Ujiko combines the power of Flash and PHP to create a search engine that, like an elephant, never forgets.

*The Flash SharedObject offers functionality very similar to cookies by storing information on the user's local hard drive. However, the SharedObject is much more powerful and can store up to 100KB of information. Like cookies, though, the SharedObject can be disabled by the user. You'll use the SharedObject to store a player's score in Chapter 7.*

Arguments of almost religious intensity rage over whether one server-side technology is superior to the rest, but I'm not going to be drawn down that path. Each has its merits and drawbacks, and all do just about the same with equal efficiency. You have to choose one, and the choice for this book is PHP in conjunction with the Apache web server and the MySQL relational database system.

## What PHP, Apache, and MySQL have to offer

Several factors stand behind the decision to choose this particular combination:

- **Cost:** They're free. While this is an obvious attraction to individuals on a limited budget, don't be put off by the thought that "you get what you pay for." The release of PHP 5 in mid-2004 brought full object-oriented capability to the language, as well as greatly improved methods of handling XML. MySQL is used by many leading organizations, including NASA, the U.S. Census Bureau, Yahoo!, and the New York Stock Exchange. The fact that more than two out of every three web servers run on Apache speaks for itself.
- **Open source:** As open source technologies, all three benefit from a rapid upgrade policy based on need rather than commercial pressures. If a bug or security risk is identified, the input of many volunteers helps the core development teams solve any problems rapidly. Future versions are available for beta testing by anybody who wants to take part, and they aren't declared stable until they really are. The same thriving community offers assistance and advice to newcomers and experienced programmers alike.
- **Cross-platform capability:** PHP, Apache, and MySQL all work on Windows, Linux, and Mac OS X. You can develop on your personal computer and deploy exactly the same code on the production server, even if it's running on a different operating system.
- **Security:** Although it is impossible to predict future developments, Apache servers are rarely targeted by virus attacks. Sensitive content can also be stored more securely in a database, with access restricted through PHP session control (this is the subject of Chapter 9).
- **Widespread use:** Both PHP and Apache are the most widely used technologies in their respective spheres. A regular survey by Netcraft ([http://news.netcraft.com/archives/web\\_server\\_survey.html](http://news.netcraft.com/archives/web_server_survey.html)) shows that Apache has consistently maintained a market share in excess of 60% of all web servers. In November 2004, it stood at more than 67%—more than three times that of Windows-based servers. PHP availability seems to march on ever upward; in late 2004, it was in use on nearly 17 million domains. And according to the MySQL website ([www.mysql.com](http://www.mysql.com)), MySQL is the world's most popular open source database, with more than 5 million active installations in late 2004.

- [download online \*The Universe in the Rearview Mirror: How Hidden Symmetries Shape Reality\*](#)
- [read online \*Topology of Surfaces \(Undergraduate Texts in Mathematics\)\*](#)
- [Havana Red here](#)
- [Danube: A Sentimental Journey from the Source to the Black Sea pdf, azw \(kindle\), epub, doc, mobi](#)
- [Neuroradiology Signs book](#)
- [download online \*Paradoxes \(Polity Key Concepts in Philosophy\)\*](#)
  
- <http://thewun.org/?library/The-Universe-in-the-Rearview-Mirror--How-Hidden-Symmetries-Shape-Reality.pdf>
- <http://fitnessfatale.com/freebooks/The-Foundation-of-the-Unconscious--Schelling--Freud-and-the-Birth-of-the-Modern-Psyche.pdf>
- <http://twilightblogs.com/library/I-Stooged-to-Conquer--The-Autobiography-of-the-Leader-of-the-Three-Stooges.pdf>
- <http://reseauplatoparis.com/library/An-Anthology-of-Philosophy-in-Persia--Volume-2--Ismaili-Thought-in-the-Classical-Age.pdf>
- <http://test1.batsinbelfries.com/ebooks/Neuroradiology-Signs.pdf>
- <http://twilightblogs.com/library/Paradoxes--Polity-Key-Concepts-in-Philosophy-.pdf>