

DEVELOPING WEB APPLICATIONS

with **NETTALK**

CAPESOFT SOFTWARE

by Bruce Johnson

authentication
ssl
browser
web
database
http
javascript
css
ajax
back
templates
styles
unicode
rad
xml
client side
html
shtml
asynchronous
dom
url
server
client
security
cla

4th edition



Table of Contents

section 1 - Getting Started	8
Introduction.....	9
The Ultra-Rapid Introduction to Web Applications	12
Creating a Simple Web Application (by Wizard)	17
Connecting to a Running Web Server	24
A Quick Look at the Wizarded App	25
Deployment.....	27
Pages, Controls & Source	31
PageHeaderTag and PageFooterTag	32
Adding Menu(s).....	34
Content Body.....	40
Popups.....	42
Template Expressions.....	45
Variables	47
Adding a Browse.....	50
Adding a Form	55
Adding a Report.....	59
Ajax	68
Lookups	69
Auto-Complete	73
Local or Global Queues (Web Apps).....	76
A CSS Primer	78
Themes	80
End of Getting Started.....	82
section 2 - Serving.....	84
Serving Static Web Sites	85
Serving Static Sites with Dynamic Content	86
Serving PHP Pages	89
Customizing the Error Page.....	91
Serving Files.....	93
Multi-Tab support	97
Sending Emails from a Web Server	99
Performance Considerations.....	103
Thread Pools.....	107
An Introduction to the Log	109
Debugging Using Debugview++.....	112
section 3 - WebServer Procedure	114

Basic Window	115
Logging Control Template	117
Performance Control Template.....	118
Settings Control Template.....	119
section 4 - Browsers	122
Browse Filters	123
Sorting Browsers.....	127
Searching Browsers (Locators)	130
Using a Form as the Criteria for a Browse.....	133
Browse Layout.....	136
Browse CSS Tips.....	140
Browse Anchors.....	143
Edit / Insert in Place	146
Calling a Form From a Browse.....	148
Browse Children	151
Combined Browse and Form.....	153
section 5 - Forms.....	156
Basic Form Setup.....	157
Form Layout.....	160
Tab Style	163
Form Fields	166
Form Field Types	168
Dynamic Forms.....	176
Popup Forms	181
Priming Fields	182
Forms Without Browsers.....	184
Accepting Incoming Files (aka File Uploading).....	186
Server-Side Processing (aka Start Buttons).....	190
section 6 - Calendars	194
NetWebYear – “Months” Layout	195
NetWebYear – “Planner” Layout (Vertical).....	202
NetWebYear – “Planner” Layout (Horizontal)	207
section 7 - Small Screen	208
What is a Mobile App?	209
Farewell jQuery Mobile.....	211
What is a Small Screen Device?.....	212
Data Driven Design versus Task Driven Design	214

How to Detect the Browser is on a Small Screen?	216
Changing Your Application	217
section 8 - Security.....	222
Introduction.....	223
Certificates.....	224
Let's Encrypt	227
Serving TLS Pages	231
External Security.....	236
Logins & Levels	237
Storing Passwords	244
Safe and Unsafe HTML	246
section 9 - Web Sockets	248
Introduction.....	249
Enabling Web Socket Support.....	251
Using WebSocket Connections (Template Features).....	252
Custom Message Processing.....	256
Section 10 - Creative Serving.....	260
Basic Site Layout.....	261
Creating and Serving Graphs.....	263
Creating and Serving a Captcha	265
Creating an Excel or Word Document (OI).....	267
Export Button - Creating a Spreadsheet (xFiles)	268
Translating Text at Compile Time.....	269
Translating Text at Runtime	272
Foreign Character Sets (aka Unicode).....	274
How IP Addresses Work	276
Using the Multi-Site-Host.....	278
Busy Indicator.....	286
Timers.....	288
Interacting with other Servers	290
Temporary Tables.....	291
section 11 - CSS and JavaScript.....	294
Cascading Style Sheets (CSS)	295
Designing Your CSS Visually.....	301
Visual Design using Themer (86).....	304
Google Fonts.....	309
JavaScript.....	312

jQuery Widget Options.....	317
section 12 - API Server	321
Introduction.....	322
Alphabet Soup	325
Creating an API server using NetTalk	330
NetWeb Server Wizard.....	331
API Documentation	340
NetWebService Procedures	347
NetWebServiceMethod Procedures	349
REST and SOAP	357
Deployment.....	359
Authentication.....	360
section 13 - Disconnected Apps.....	366
Introduction.....	367
Database Design for Distributed Data.....	369
API Sync Server	377
Disconnected Windows Desktop Apps	380
Disconnected Web Apps	382
section 14 - Webserver Inside	387
What is p_web?.....	388
Incoming Request Properties	391
Procedures Are Event Handlers	393
Code Flow inside the WebHandler.....	394
Understanding Generated Code	399
Advanced Page Dynamics.....	404
Understanding the Value Queue.....	406
Manipulating the Session	408
Manipulating the Session Data	410
Deleting a Session.....	412
Epilogue.....	413
Acknowledgements.....	414
Section 15- Appendices.....	415
Appendix A – Events & Stages.....	416
Appendix B: WebHandler Common Embed Points	419
Appendix C: Common p_web Properties	420
Appendix D: NetWebBrowse Quick Reference	422
Appendix E: NetWebForm Quick Reference	424

Appendix F: Useful built-in CSS styles 426

section 1 -

Getting Started

The NetTalk templates come with the ability to create a web server. The goal of this first section in the book is to introduce you to some of the fundamental concepts of building your first web server.

Introduction



Tip: The Web is a different country, and they do things differently there. Items of special interest will be marked like this to draw your attention to important concepts.



Security Note: The web is a dangerous place, populated with lots of people trying to make your server do bad things. When you do things right it's very safe, but when you do things wrong bad things can happen to nice computers. NetTalk is very secure by default, however you should pay special attention to notes marked like this. Avoid these notes at your own peril.



Lightbulb: Sometimes small facts, properly understood, can make things much easier to understand. These facts are mentioned in a box like this. Take a moment to digest the contents.

This book was first written as the teaching aid for a two-day NetTalk Web Server training course run in 2010. The goal of this book is not so much to cover every eventuality, or list every template option, but rather to cover a wide range of topics that affect the building of a web application using Clarion and NetTalk.

While in places it overlaps with the shipping documentation, it is not meant to be a replacement for that documentation. The shipping documentation remains a key aspect of learning NetTalk and using it to its full potential. In places, this book will refer you back to the documentation for a more in-depth discussion of a specific topic. While the documentation is more referential, this book is designed to be more tutorial in nature. It does however contain information that will be useful regardless of your level. Rereading the book from time to time is recommended.

The topics in the book start with chapters aimed at beginners and get progressively more advanced as it continues. It covers the whole life cycle of the application, from creation to improvement, from debugging to deploying.

There are many important differences between programming for a desktop environment, and programming for the web. The learning curve for this transition can be steep. It is therefore recommended that you work through this text slowly, absorbing each step before moving onto the next one. Going back and re-reading sections is also an important way of reinforcing the knowledge. On each pass through the text more detail becomes clear, because you are able to understand it as part of a larger framework.

My thanks go to all the programmers out there who are using NetTalk to build web applications, and who continue to feed back to me, on an almost daily basis, their suggestions, feature requests and of course bug reports. It is thanks to their efforts that NetTalk is where it is today.

Sites versus Applications

The web is a large space, full of interesting places, some useful, some dangerous. It started out purely as a way to publish, and consume information. In many ways that is still its prime purpose. However,

communication is a two-way street, and while communicating with customers and users, is important, it's a lot more interesting when they can communicate back with you. Data consumed by users becomes vastly more useful when that data is created by users as well.

In this regard, it's useful to discern the difference between a web *site* and a web *application*. A web site is primarily (although not necessarily exclusively) for the presentation and publication of data. It is easily searched by engines such as Google and users can consume it in a largely one-way direction. Web applications require so much more, they act as a place where people can not only consume data, but consume data specifically directed at themselves. They can modify that data, add to it, tweak it, share it with others and, yes, consume it as well.

Wikipedia is an example of this sort of approach. Early online-encyclopedia sites focused on the presentation of data they collected, articles they authored. They were sites in the truest sense of the word. Wikipedia envisaged a web application, one where users could not just consume content, but create it as well. I would suggest that their experiment has forever changed how we see encyclopedias. The biggest sites on the web today, sites like Facebook, Google, Twitter and many more are focused not on user consumption, but on user participation.

Where most web tools are focused on creating sites, with some interaction, NetTalk was built primarily for interaction. With NetTalk the presentation is not the first item to consider, first comes function. It is easy to change the appearance of the well-functioning application; it is less easy to tack-on function to a design primarily aimed at only consumption of data.

Don't get me wrong – presentation is import. And we'll cover that. But unlike a Windows application the appearance of the application and its functionality are completely separate. Thus the work revolves around the function, the appearance can be handled later.

API Servers / Web Services

In recent years a rise in a third kind of web application has occurred. This is a web service application, an application, hosted on the web, but which is aimed not at humans but at other computers. It is a way for programs to communicate with each other. They are all around us, but in most cases invisible. For example, when your phone sends an image to your cloud storage, it is using an API to do that.

Although Web Services are a more recent addition to the web lexicon, they are much simpler to make than a web app (which has to create a human-readable interface.) Equally creating a Web Services server in NetTalk is trivially easy compared to creating a Web Application server. If your primary reason for building a NetTalk app is to provide a Web Service application, then you are in for a treat.

Fourth Edition

The first edition of this book was current to the end of 2010, the second to the end of 2013, the third to the end of 2017. However NetTalk has continued development since then. The goal of this edition is to add material to cover the new NetTalk features added during 2017 to 2022, including features added to NetTalk 11 and NetTalk 12. This book assumes you are using NetTalk 12 (or later) so some of the options and features may be missing if you are using an earlier version.

Lies, Lies and Tutorials

During the course of this book I will make many statements. Some of them are true. Some of them are not. In order to present ideas and concepts in a digestible way it is sometimes necessary to bend the absolute truth to something which is easier to grasp. In the same way we were taught that the world is

round (it isn't) or that we only have five senses (we don't) it can be easier to present the broad truth without going into the exceptions to the truth.

Where possible I will try and make a note of things which are factually wrong, and expound on the idea in a footnote.

What to Learn First?

Web Apps

NetTalk contains a lot of options. And the web does things in many possible different ways. Sometimes when working through this book you'll encounter things that aren't explained at that point. Try and take them on faith for now and hopefully more understanding of the step will be revealed later on.

This book is designed to be read in a Tutorial way – several topics are covered in a basic form to begin with, and then in a more complex form later on. So working through the book from cover to cover is the recommended approach if you are building web applications.

Web Services

If however your primary goal is to create a Web Services application, then there are some short-cuts you can take. I recommend covering;

- a) The first 8 chapters of Section 1 – to the end of the chapter on Deployment.
- b) Section 3, Section 8 and Section 12.