

# **Building Web Applications Using emPOWERnet 2.5**

Reference for web developers



Park City Solutions, Inc. has prepared this document for use by Park City Solutions personnel, licensees, and customers. The information contained herein is the property of Park City Solutions and shall not be copied, photocopied, translated, or reduced to any electronic or machine readable form, either in whole or in part, without prior written approval from Park City Solutions.

© 2004 by Park City Solutions, Inc. All Rights Reserved. Printed in the USA..

**Trademarks and registered names.** emPOWERnet is a trademark of Park City Solutions, Inc.

Microsoft is a registered trademark of the Microsoft Corporation.

IBM is a registered trademark of IBM Corporation.

Oracle is a registered trademark of Oracle Corporation.

eWebEditPro is a trademark of Ektron, Inc.

All other trademarks or registered trademarks are the property of their respective owners.

Park City Solutions, Inc.  
5755 Granger Road  
Independence, Ohio 44131-1461  
Main 216.485.4600 Fax 216.485.4606  
[www.parkcitysolutions.com](http://www.parkcitysolutions.com)

# Table of Contents

Chapter 1: Introduction ..... 4

    Before You Begin Development..... 4

        Skill and Experience Prerequisites..... 4

        Installation and Site Setup ..... 7

    Why emPOWERnet? ..... 7

# Chapter 1: Introduction

Park City Solutions is pleased that your institution has decided to use the emPOWERnet™ 2.5 Toolkit and Framework. With this combination of tools, you can quickly and effectively develop Web-based applications and services.

emPOWERnet features an XML-based document store which serves as the core repository for both the Web Authoring Center and portal modules, including all the XML documents holding specific categories of information.

emPOWERnet offers you a powerful set of efficient, easy-to-use development tools with which to create portal solutions for applications such as patient management, newsletters, frequently asked questions (FAQs), staff directories, policies and procedures guides, purchase order requisition forms, and stock applications.

This guide provides reference information about the tools and the development environment, including the tasks to complete the following steps:

- Produce well-formed XML documents to generate forms, views, and navigational paths that comprise a web module
- Use directional keyword searches and approval routing workflows to add value to a module
- Publish, view and test a module on the WAC
- Publish the completed module to a web site.

Tips and some best practices are also included.

## *Before You Begin Development*

### *Skill and Experience Prerequisites*

Experience with at least one of the following development tools should provide you with a platform from which to understand and use the emPOWERnet Toolkit:

- XML and XSLT
- HTML
- VB Script or JavaScript
- Visual Basic [COM objects]
- Active Server Pages [ASP]
- C++ or Java

Experience using the following business skills should also be helpful:

- Front end-back end IT/Application integration
- Business Analysis/User Interface Design
- Project Management

## *Installation and Site Setup*

You must complete the following tasks before you begin developing modules with the emPOWERnet Toolkit.

1. Install and test the hardware that meets minimum specifications for emPOWERnet.
2. Install SQL Server 2000 [service pack 3] or Oracle 9i [version 2 recommended] and the most current service packs. Make certain that you have invoked the full-text indexing option when installing SQL; full-text indexing is handled in the background for a normal Oracle installation.
3. Install the emPOWERnet Framework and Toolkit.
4. Launch and log onto the Web Authoring Center using the Administration account.
5. Connect to an existing emPOWERnet web site.
6. Review the document entitled "Site Administrator's Guide for the emPOWERnet Web Authoring Center" that was supplied with your portal module. This document explains in detail how one or more designated functional users serving in a site administrator role will use the modules you build for the Web Authoring Center to create, organize and maintain web site content for your institution.

## *Why emPOWERnet?*

The emPOWERnet Toolkit consists of a set of tools for the rapid development of web modules so that administrators and content providers can add and manipulate content for their organization's Web site.

Using the Toolkit, you can build customized portal modules for content authors who design, create collateral for and manage the complete content lifecycle, including reviews, approvals, and delivery. Your delivery platform can span internet, intranet, and extranet sites, as well as other channels including multimedia and wireless applications.

XML, an SGML-based extensible markup language, is used to create an unstructured document store with the versatility to hold many types of information. Leveraging a component-based architecture with loosely-bound structures in an extensible framework environment gives you the flexibility to keep your development efforts simple and straightforward, and to create custom components to extend their usefulness.

Because of the flexibility and versatility of the emPOWERnet development environment, there is no need to have the exact contents of a document in hand when you design an application. Further, controls can be easily added or removed from a form--without affecting the application.

The benefits of XML and the emPOWERnet Toolkit for both you and your organization are many, including:

- Wizards that guide you through the process of building well-formed XML documents, along with utilities for user management
- Automatic normalization of incoming data to XML through the emPOWERnet data services layer
- Data manipulation consistency, with low-level service handling by the application framework, freeing you to focus on user deliverables
- Familiar development tools including COM+ and MMC
- N-Tiered technology for support of business applications
- Scalability for a high volume of transactions, activities and users with minimal use of resources
- Efficient maintenance of security and authorizations, with responsibility shared with your content managers
- Resiliency for deployment in redundant, high-availability data centers
- Platform flexibility to integrate your modules into existing solutions such as IBM® Websphere or Microsoft® Sharepoint through ASP
- Built-in management of Visual Basic components
- Control of the “look and feel” of your portal by your institution.

An XML document arranges information in a hierarchy that allows human or machine readers to understand the patterns in the information, and to access them selectively. With XML you can:

- Separate logical content of a document from its presentation
- Readily synthesize data from disparate sources into different packages, depending upon the needs of the user
- Establish a shared context between document creator and document receiver by using a document type definition and tags to define metadata
- More readily rebuild a table in the database, should it be dropped.

In addition, you are free of the need to deal directly with content and relational database tables. With XML, there is an increased emphasis on well-formed documents and comprehensive indexing.

The following table identifies specific performance and output comparisons between client-server and the XML development environments using the emPOWERnet Framework.

<b>Management</b>	<b>Client Server</b>	<b>XML with the emPOWERnet Framework</b>
Data Storage	Data is stored in relational tables	Data is stored in hierarchical XML documents
Indexes	RDBMS Table Indexes	Indexes denormalize the data, support forms, views, and searching
Presentation Formatting	Instructions are located within the code, often co-mingled with the business logic	Instructions are stored in XSL files
Data Access	Application manages data access	Framework manages data access; there are fewer database access calls
Full-text searching	Application manages a full-text search	Framework manages a full-text search