

# Architecture Services Documentation Standards

**Doc. Version 0.3  
18-July-2003**

The controlled master of this document is available online. Hard copies of this document are for information only and are not subject to document control.

## Amendment History

Project Name: *Architecture Services Documentation Standards*

Amendment History—Document Status (e.g. Draft, Final, Release #): **Draft 0.3**

CR# (optional)	Document Version #	Approval Date	Modified By	Section, Page(s)and Text Revised
	V0.1	Jan.-May/2003	Lauri McIntosh	All (creation of original document)
	V0.2	06/16/2003	Lauri McIntosh	<ul style="list-style-type: none"> <li>▪ Cleaned up/edited for use outside Rebar team.</li> <li>▪ Put content into Rebar guide format (following EIT and current Rebar documentation format standards); added introductory and summary chapters.</li> </ul>
	V0.3	07/18/2003	Lauri McIntosh	Continued work from V0.2 above.

# Table of Contents

## *Architecture Services Documentation Standards*

Amendment History.....	ii
<b>Introduction: <i>Architecture Services Documentation Standards</i>.....</b>	<b>v</b>
Purpose .....	v
Audience.....	v
Prerequisites .....	v
<b>Chapter 1: Overall Guidelines and References.....</b>	<b>1</b>
Introduction .....	1
Standards Hierarchy.....	1
Two Critical Documentation Requirements .....	2
When to Use This Guide.....	2
Support—Template and Examples .....	2
<b>Chapter 2: Terms and Usage.....</b>	<b>3</b>
Introduction .....	3
General Terms .....	3
<b>Chapter 3: Fonts.....</b>	<b>7</b>
Introduction .....	7
Boldface.....	7
Underlining.....	7
Quotation Marks .....	8
Italics .....	8
Capitalization.....	9
<b>Chapter 4: Document Format, Elements, and Organization.....</b>	<b>10</b>
Introduction .....	10
Headings .....	11
Page Headers and Footers.....	12
Standard Introductory Pages.....	14
Introductory Page Format .....	14
Introduction “Chapter” .....	14
Main Content Chapters (Body).....	17
Tables.....	18
Example 1—If-Then Table .....	18
Example 2—If-Then Table .....	18
Example 3—If-Then Table .....	18
Example 4—If-And-Then Table .....	19
Example 5—Time-to-Complete-Documents Table.....	19
Table Format:.....	20
Table Wording and Text Styles:.....	20
Graphics and Screen Shots (all OH style):.....	21
How to Organize Content .....	23
<b>Chapter 5: To Do Before Finalizing Document.....</b>	<b>24</b>
Introduction .....	24
Cleanup.....	24
Prep for Review or Release .....	25

<b>Summary and Help</b> .....	<b>26</b>
Guide Summary .....	26
Support—Template and Examples .....	26
Contact Us .....	26

---

## Introduction: *Architecture Services Documentation Standards*

---

**Purpose** This guide documents the **standards, formatting, and style guidelines** you should use when creating or editing Architecture Services documentation.

---

**Audience** The following people make up the audience for the *Architecture Services Documentation Standards*:

- **Developers** or **writers** creating documentation for Architecture Services
- **Editors** or other **reviewers** (such as conformance review participants) who must review Architecture Services documentation

### Prerequisites

There are no prerequisites for using *Architecture Services Documentation Standards*.

## Chapter 1: Overall Guidelines and References

---

### Introduction

This chapter covers the following **overall information** about the Architecture Services documentation standards:

- Standards hierarchy
  - The two critical requirements for documentation
  - When to use this standards guide
  - How to find help when using these standards
- 

### Standards Hierarchy

Architecture Services documentation should follow these standards, **in order of importance**:

1. Standard English grammar, spelling, punctuation, and usage rules.
2. Standard English style and usage rules, as governed by the latest editions of the following references:
  - *EDS Style Guide*
  - *Associated Press Style Book and Libel Manual* (The *EDS Style Guide* is based heavily on this publication and is organized in exactly the same way. The *EDS Style Guide* lists the AP guide as a main reference.)
  - *The American Heritage Dictionary* (college edition) or *Webster's New World Dictionary* (college edition)
  - *The Elements of Style* (by William Strunk, Jr., and E.B. White)
  - *The Chicago Manual of Style*

Note: Except for the *EDS Style Guide*, the references above have been the communication-related standards for decades. You will find them listed as main references in the *EDS Style Guide* and the *Associated Press Style Book and Libel Manual*, which is one of the standard references itself. They do not change; they only add new terms based on new or evolving industries, such as the computer industry. EDS Corporate Communications and other large EDS groups have also used these references for many years.

3. Information Mapping-based format standards, as used by EDS since the early 1990s.
4. [Microsoft Manual of Style](http://www.microsoft.com/downloads/details.aspx?FamilyID=b494d46b-073f-46b0-b12f-39c8e870517a&DisplayLang=en) standards when they do not contradict the above standards. Use this guide mainly to determine the correct spelling, capitalization, and punctuation of technical or Microsoft **terms**—not for formatting or style standards. (See the following URL:  
<http://www.microsoft.com/downloads/details.aspx?FamilyID=b494d46b-073f-46b0-b12f-39c8e870517a&DisplayLang=en>)

---

**Two Critical Documentation Requirements**

All standards, however, must be implemented while maintaining the following two critical requirements of documentation:

- Technical accuracy
- Clear communication to the user (based on the documentation medium and the user's experience and skill level).

These two requirements supercede all standards; however, in most cases the standards will also support accuracy and communication since they were designed with that goal in mind.

---

**When to Use This Guide**

If you are writing or editing documentation, you should use this guide at least twice:

- Read this guide before you begin creating/editing documentation so that you know the terms and areas that the standards affect.
  - Refer to this guide while you create/edit documentation.
- 

**Support—Template and Examples**

There are two ways to get support when using the Architecture Services documentation standards:

- **Arch\_Services\_DocStandards\_Template.dot** MS Word template. (This template contains some Rebar-specific content but can easily be modified.)



Arch\_Services\_DocStandards\_Template.doc

- **Examples of documentation using this format**—You can see finished documentation created in this format on line at <http://eads.eit.edu/EADS/development/rebar/doc.aspx>:
  - *Rebar ADO.NET Data Access Guide*
  - *Rebar QuickPort Web Application Deployment Guide*
  - *Rebar Config.xml Guide*
  - *Rebar .NET Exception Handling Guide*
  - *Rebar QuickPort Developer's Guide*
  - *Rebar QuickPort Installation Guide*
  - *Rebar Prerequisite Software Installation Guide*



## Chapter 2: Terms and Usage

---

### Introduction

This chapter documents the correct use and style for terms and phrases common to Architecture Services documentation. As more terms are discovered and standardized, writers/editors should update this list. This list covers many issues; here are a few examples:

- Correct/incorrect terms and phrases—what is the correct or standard way to refer to a screen element, an action, etc.?
- Capitalization—is a certain term all caps, title case, lower case, “camel case,” or something else? Sometimes there is no standard, so for consistency, a project team must make the decision and add it to this list. You must also look at the owner of a term; for example, “.NET Framework” is a Microsoft term, but “Rebar framework” is not, so you follow the standard set by the “owner.”
- One word versus two words
- How to word and format examples, bulleted lists, steps, etc.

This list is in **alphabetical order**; some terms appear in more than one category.

---

### General Terms

- **actions** (buttons to click, items to check, items to look for/highlight, CDs to insert, etc.)—Bold the name itself but not the generic term; example “Click the **Close** button.”
- **appear**—Say that something “will appear,” not “should appear.”
- **ASP**
- **ASP.NET**
- “**at the window/dialog box...**”—Use “in” instead.
- **buttons**—Add “button” to button names—“click the **Cancel** button,” not just “click Cancel.” Initial cap (title case) the name of the button but not the generic screen element name. See also “radio button.”
- **Caution:**—Use this term when you need a strong warning that the user must notice. Use initial caps, bold, underlined font and follow with a colon and two spaces (bold the colon but don’t underline it). This word has two special effects applied to it (bold and underline) because it is used when significant consequences can occur if the user does not heed its warning.
- **CD titles**—Italicize the proper name/title of a CD since it is a publication (but not generic terms that are not part of the title); example *Visual Studio .NET Component Update* disk (notice “disk” it not italicized or initial capped). Do not use italics when referring to a non-proper name/title of a CD, such as “You will be asked to insert Visual Studio .NET disks 2, 3, and 4” or “Visual Studio .NET installation CDs.” (In other words, follow standard English punctuation/format rules for titles, just as you would italicize a music CD “album” title but use quotation marks around titles of individual songs on the CD.)
- **check boxes**—Be sure to include “check” when referring to these; do not just call them “box” so that they are not confused with dialog boxes.

- **code example**—Once one of the code example styles is applied in MS Word, highlight the entire code block and move its left margin two “notches” (Word ruler notches; ½ inch) to the right of the left margin of the text directly above it. Be sure to end it with one empty Normal-style paragraph (this gives sufficient room between it and the next block of text). Bold relevant parts of the code—the main points you refer to in the explanatory text that accompanies it.
  - Font—Courier 9pt black (**Code example** style for blocks of code; **Code--in text** style for code mentioned within step text.)
  - You can make the code example’s font smaller (down to 8 point) if needed to get all lines of a section of code on the same page as the instructions/steps describing it (or to keep a code line from wrapping).
- **config.xml**—Lowercase, unless using “XML” by itself.
- **contractions**—Do not use them.
- **Data Layer Builder**—Call it the “EDS Data Layer Builder” instead of the “Rebar Data Layer Builder.”
- **dialog box**—Do not leave out “dialog.”
  - “**in the dialog box**, click/choose/enter...”—Use this wording instead of “on the dialog box...” or “at the dialog box...”
  - **dialog box** vs. **window**—Define each and then revise doc. See also “window.”
  - “**at the window/dialog box...**”—Use “in” instead.
- **EDS Data Layer Builder**—Call it the “EDS Data Layer Builder” instead of the “Rebar Data Layer Builder.”
- **enter** vs. **type**—Use “enter xxx” instead of “type xxx” when showing how to fill out fields, etc.
- **examples:**
  - **Within text**—Use “(example: edogs)”, with “example” lower case, not bold; colon and one space following; the example itself in bold; put all in parentheses.
  - **Code examples**—Use “Example:” (or “Example—Providing a custom encryption key:” if you need to add more info to example line) on a line (**txt style**) above the code example (use colon at end of “Example” line; only cap the first word after the dash if a dash is used).
  - **Subsections (h5 style)**—Use this format: “Example—Standard Attributes for Application Config.xml File” (dash after “Example”; no colon at end of line; title case for capitalization).
  - **Example code outside of steps** (or in reference-guide style document)—Use “**Example:**” in bold on a separate **txt** style line above the code block.
- **file path examples**—Use Microsoft Sans Serif 10pt bold font; **file path example** style.
- **framework**—Initial capped when used with .NET: “...the Microsoft .NET Framework.” NOT capped when used with Rebar: “...the Rebar framework.”
- **generic screen element names**—Do not initial cap with their proper names; example: Start menu; OK button.
- “**in Visual Studio**”—Be consistent within a document with “**In the Visual Studio window**” vs. “In Visual Studio.”
- **link**—Do not call an underlined choice a button—call it a “link.”

- **lists**
  - **step lists**—Do not use “Steps:” headers before step lists.
  - **substeps under a main step** (usually **bu2** or **bu2-last** style)—If the user is filling in fields, bold the field name, follow with a dash, then begin action in lower case.  
Example:  
**Project Types field**—click rebar Projects.
  - **terms list** (bulleted) with description/explanation following:
    - Bold the term (before the dash)
    - Begin the explanation (following the dash) by initial capping the first word.
    - Example:  
**Message property**—The **BaseApplicationException object** that is built into the Exception Management Application Block **does not override the Message property** like the EDS.EITCommon.Exception object does.
- **menu item progressive choices**—Example: “In Visual Studio, choose the **File>Add Project>New Project...** menu item” (no underline; bold the menu names; follow with ellipsis if that actually appears in the menu name; use this ► arrow character.
- **.NET**—Not “.Net”.
  - “**the .NET run-time service**”—Do not use “.NET run-time” or “the .NET run-time.”
- **page**—Do not use it to mean “window,” “dialog box,” or even “screen. Only use when referring to a web page.
- **popup**—Not pop-up (Search “pop-up.”)
- **programmatically**—Spelled with double “m.” (This is consistent with MSDN Library usage.)
- **radio button**—Refer to it as a “radio button,” or just tell the user to “select xxx.” See also “button.”
- **Rebar**—Initial capped
- **Rebar Data Layer Builder**—Do not call it this; use the “**EDS Data Layer Builder**” instead of the “Rebar Data Layer Builder.”
- **right-click**—Hyphenated.
- **run-time**—Hyphenated; no initial capped “run-time.”
- **screen**—Do not use it to mean “dialog box” or “window.”
- **screen element names**—Do not initial cap generic screen element names with their proper titles; example: Start menu; OK button.
- **Session state**—Initial cap “Session” but not “state.”
- “**the .NET run-time service**”—Not “.NET run-time” or “the .NET run-time.”
- **VB.NET**—Use all caps when abbreviated like this.
- **Visual Studio .NET**—Notice spaces, capitalization.
  - “**in Visual Studio**”—Be consistent within a document with “**In the Visual Studio window**” vs. “In Visual Studio.”
- “**will appear**”—Say that something “will appear,” not “should appear.”

- **window**—The following are considered windows (notice that some are not capitalized):
  - Add/Remove Programs
  - browser
  - code
  - Command Prompt
  - Internet Information Services
  - Solution Explorer (within Visual Studio)
  - SourceSafe
  - Visual SourceSafe
  - Visual Studio .NET Setup
  - Visual Studio .NET Setup Options
  - Visual Studio .NET Setup Start
  - Visual Studio code
  - Visual Studio, Output (within Visual Studio)
  - Windows Components Wizard
  - Windows Update
- **“at the window/dialog box...”**—Use “in” instead.
- **XML**—All caps when used by itself; lowercase when using with a term, such as “config.xml”; see also “config.xml.”

## Chapter 3: Fonts

---

### Introduction

This chapter defines the standards for formatting text:

- Boldface
  - Underlining
  - Quotation marks
  - Italics
  - Capitalization
- 

### Boldface

Note: Use as boldface sparingly as possible and not in all paragraphs.

- Use in text and steps for the following:
    - Buttons to click
    - Actions to take
    - Menu to click
    - Menu item to choose
    - Main point of a significant paragraph (to be sure user does not miss it)
    - Main subject word of each bullet in a bulleted list that contains more than just terms
    - Parts of code examples that text references
    - The word “**Caution:**” when it is used as a strong warning that the user must notice. Use initial caps, bold, underlined font and follow with a colon and two spaces (bold the colon but don’t underline it).
  - All headings are bold:
    - Chapter titles (**H3** style)
    - Section titles (**h4** style in left margin)
    - Subsection titles (**h5** style in text)
    - Sub-subsection titles (**h6** style in text)
- 

### Underlining

Note: Use underlining as sparingly as possible. Use in text and steps for the following:

- Almost any use of “not” (readers’ eyes often do not see this word).
- Extremely significant words (such as “**Caution:**” “required,” “Note:” etc. that it is critical that the user sees). The “**Caution:**” will also be bold (use initial caps, bold, underlined font and follow with a colon and two spaces; bold the colon but don’t underline it).
- Hyperlinks—Use for cross-reference links within the document (such as “See also” page and section/subsection references), for email addresses, and for URLs.

---

**Quotation Marks**

Use Standard English and *EDS Style Guide* rules for quotation marks (quotes, words referred to as themselves, article or other composition titles, etc.). Use in text and steps for the following:

- Section and subsection titles when referenced in text.  
**Example:** If you do not supply this value, it defaults to “false”.

**Exception**—Follow the guidelines above except when putting quotes around terms or values to be put into **code**; in this one case, periods and commas do not go inside quote marks.

---

**Italics**

Use Standard English and *EDS Style Guide* rules for italics. Use in text and steps for the following:

- **Caution:** Use sparingly since it is notoriously hard to read, especially when viewing a document online.
- Do not use italics for emphasis; use underlining instead, and use it sparingly. (See “[Underlining](#)” subsection above on page 7.)
- Publication titles, including CD titles and official document names (example: *Rebar Config.xml Guide*, *Rebar QuickPort Developer’s Guide*).
- Page headers.
- **h6** style headings
- Placeholders in code examples—These placeholders represent information that the user or implementation must provide. Example:

*drive:\Program Files\Common Files\EDS\EIT\applicationAcronym*

(In this path, “drive” is a hard drive letter such as C or D, and “applicationAcronym” is a unique identifier for the project.)

---

**Capitalization** Use Standard English and *EDS Style Guide* rules for capitalization.

- **keys and key combinations**—All caps (use “+” for key combinations); example: ENTER, CTRL+F11
- **screen/window element names**—Initial cap the name of the button but not the generic screen element name; example: Start menu; OK button.
- **button names**—Add “button” to button names—“click the **Cancel** button,” not just “click Cancel.” Initial cap (title case) the name of the button but not the generic screen element name. See also “radio button.”
- **menu and menu item names**—Use the same capitalization style that the menu itself uses (it is not consistent in Microsoft, so use whatever the user will actually see).

Use the following capitalization standards for headings:

- **chapter titles (H3 style)**—Title case; follow “Chapter x” with colon and two spaces; example: “Chapter 1: Requirements of the Error Trap Approach”
- **section titles (h4 style)**—Title case
- **subsection titles (h5 style)**—Title case
- **sub-subsection titles (h6 style)**—Title case

## Chapter 4: Document Format, Elements, and Organization

---

### Introduction

This chapter explains the formatting and style sheet styles used for Architecture Services documentation and covers the following:

- Headings
- Page headers and footers
- Standard introductory pages
- Introduction chapter
- Main content chapters (body)
- Tables
- Graphics and screen shots
- How to organize content

Remember that you can find detailed examples of the format shown in this document by looking at the following Rebar documentation on line at <http://eads.eit.eds.com/EADS/development/rebar/doc.aspx>:

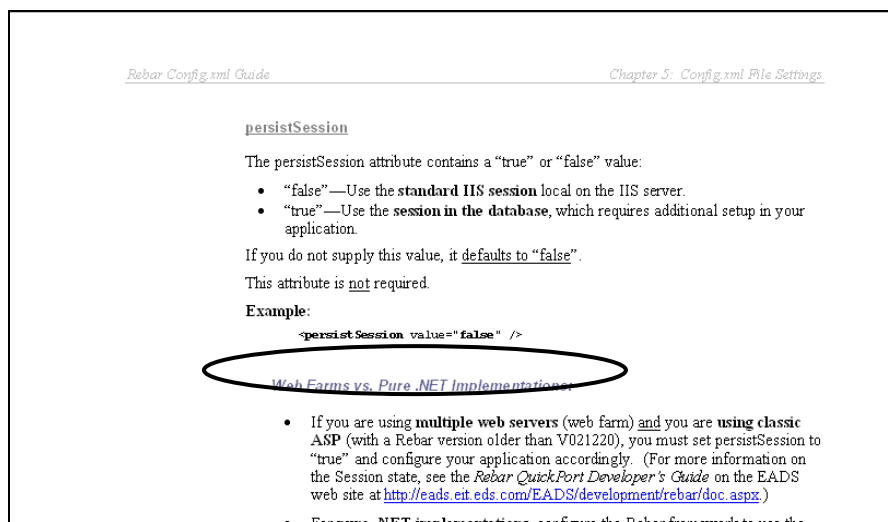
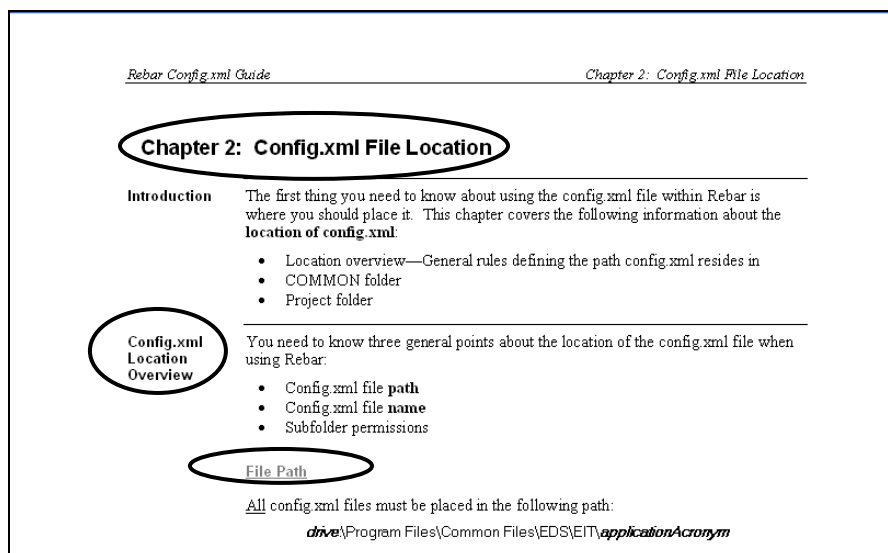
- *Rebar ADO.NET Data Access Guide*
- *Rebar QuickPort Web Application Deployment Guide*
- *Rebar Config.xml Guide*
- *Rebar .NET Exception Handling Guide*
- *Rebar QuickPort Developer's Guide*
- *Rebar QuickPort Installation Guide*
- *Rebar Prerequisite Software Installation Guide*



## Headings

The Architecture Services documentation format uses **four** headings in the body of the document (there are other styles used in the introductory pages, such as the title page):

- **chapter titles (H3 style)**—Arial 14 pt., bold, title case, flush left. (example: “Chapter 2: Config.xml File Location”)
- **section titles (h4 style)**—Arial 10 pt., bold, title case, flush left in left margin. (example: “Introduction” and “Config.xml Location Overview”)
- **subsection titles (h5 style)**—Arial 10 pt., bold, 50% black (gray), title case. (example: “File Path”)
- **sub-subsection titles (h6 style)**—Arial 10 pt., bold, italics, blue-gray, title case. (example in second graphic below: “Web Farms vs. Pure .NET Implementations”)



These four headings also appear in the automatically-generated table of contents.

**Page Headers and Footers**

Documents use two different types of headers—one for the first four types of introductory pages and another for the rest of the document.

- **Introductory page headers**—Use only on the following pages:
  - Title page
  - Statement of Confidentiality page
  - Amendment History page(s)
  - Table of Contents page(s)

Enterprise Information and Technology

Enterprise Applications  
Development Services

**Amendment History**

Project Name: *Rebar .NET Exception Handling Guide*  
 Amendment History—Document Status (e.g. Draft, Final, Release #): Release 1.2

CR# (optional)	Document Version #	Approval Date	Modified By	Section, Page(s) and Text Revised
CR 30	V0.1	03/06/2003	Luther Casey	All (creation of original document).
CR 30	V0.2	03/11/2003	Luther Casey	Updated per comments from Stephanie Bankes, David Collins, and Grace

- **Document body page headers**—Use for all other pages in the document.
  - 10 point italics (header-odd style; line is a bottom paragraph border)
  - Title of document—flush left
  - Chapter title—flush right
  - Paragraph border under paragraph; ¾ point (Format ► Borders and Shading... ► Borders tab)

Rebar Config.xml Guide

Chapter 3: Config.xml File Structure

**Chapter 3: Config.xml File Structure**

---

**Introduction**     In this chapter, you will look at the **structure** of the config.xml file, including the following characteristics:

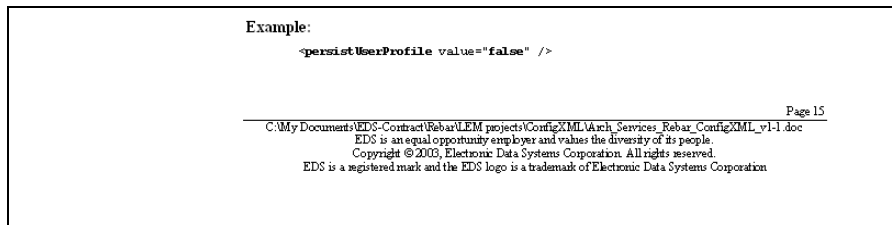
- **Footers**—Use EIT standard footer; Times New Roman 8 point.

Note: Use **Roman** numerals only for the four types of introductory pages. Use **Arabic** numerals for all others.

— Use on all pages except the title page.

— Contains the following:

- Page number
- File path
- EEO statement—“EDS is an equal opportunity employer and values the diversity of its people.”
- Copyright notice—“Copyright © 2003, Electronic Data Systems Corporation. All rights reserved.” (Note: Make sure the year is correct.)
- Trademark and logo notice—“EDS is a registered mark and the EDS logo is a trademark of Electronic Data Systems Corporation.”



**Standard Introductory Pages**

Always begin the document with these standard introductory pages:

- EIT Title page
- Statement of Confidentiality page
- Amendment History page(s)
- Table of contents page(s)

**Introductory Page Format**

The four introductory pages have different format from the rest of the document in these ways:

- **Margins**—These pages have 1-inch left and right margins; this is a smaller left margin than on the other pages in the document.
- **Page numbering**—Use **Roman-numeral** page numbering for these pages. You will use Arabic numerals for all other pages.
- **Page headers**—

**Introduction “Chapter”**

The Introduction is not the first chapter of content, but it is formatted like one—except it continues the Roman numeral page numbering from the previous standard pages.

Enterprise Information and Technology		Enterprise Applications Development Services
<b>Table of Contents</b>		
<b>Rebar Config.xml Guide</b>		
Statement of Confidentiality.....		i
Amendment History.....		ii
<b>Introduction: Rebar Config.xml Guide.....</b>		<b>vi</b>
Purpose.....		vi
Audience.....		vi
Prerequisites.....		vi
About Rebar.....		vii
Rebar.....		vii
Online Rebar Documentation.....		vii
Preparation Steps.....		viii
You Must Have.....		viii
You Must Know.....		viii
<b>Chapter 1: Config.xml Overview.....</b>		<b>1</b>
Config.xml Definitions.....		1
Comparison to Microsoft's Web Config.....		1

Include the following sections in the Introduction:

- Document’s **purpose**—Do not give an overview or introduction here; this is usually only a one- or two-sentence statement describing the overall goal of the document (similar to an objective statement for a class). Save all the other details for Chapter 1, which you can call “Chapter 1: Overview” if you need to spend more than a paragraph or two giving background or introductory information.)

**Example:** “This guide documents the steps developers must take to deploy a new Rebar web application.”

- **Audience**—List only the primary people/jobs/roles this document is written for. Do not include other people who “might” use it. Never use something like “all EDS developers” or “all EDS employees” or “anyone interested in ....” Groups this large and varied will not have the prerequisite skills/knowledge/experience to efficiently use a single document.
  - **Prerequisite** skills/knowledge/experience the user should have to efficiently use (or understand) the document.
- **“About...”**—Give any general information about the overall tool, application, topic, task, etc. that the document supports (example: “About Rebar”). Also include the following:
  - Where to find online documentation.
  - Any other related documentation, such as those that the user should use immediately before or after this one. In process-style guides (such as the *Rebar QuickPort Installation Guide*), include a **hierarchy chart of any related or prerequisite guides**.
  - Contact information for users to ask further questions or give feedback on the document’s usability
- **Preparation Steps**—List any thing the user must do before using/reading the document; include all of the following that apply:
  - You must have—Examples: access to file servers or intranet sites; specific software that must be installed; software CD(s); administrator privileges on their PC; etc.
  - You must do—Examples: prepare a specific amount of space on their hard drive; make critical Windows updates; disable a specific utility/application (such as PassKey or virus software); etc.
  - You must know—This is not prerequisite skills/knowledge/experience but project- or application-specific facts, passwords, data, etc. the user must have ready as **input** into one of the steps in the document.

**Note:** If the preparation steps are critical (in other words, being unprepared will likely cause the user to mistakes or significantly slow the him down), and you think users might skip the Introduction chapter, you may want to treat “Preparation Steps” as its own chapter immediately before Chapter 1.

- **Approximate Time to Complete...** table—List the amount of time users can expect to spend on each chapter (main task/step) of the process the document describes.

<b>Approximate Time to Complete Rebar QuickPort Web Application Deployment</b>	Chapter	Approximate time (minutes) IF all preparation tasks are done beforehand
	1 Create the Web Installer Project	10
	2 Configure the Web Installer Project	15
	3 Compile the Web Installer Project	05
	4 Deploy the Web Application Project	15
	<b>Total</b>	<b>45</b> <b>(approximately 1 hour)</b>

Note: You will only include this table if you are documenting the steps to follow in a process (such as the *Rebar QuickPort Installation Guide*); do not include this table in reference-style documentation (such as the *Rebar .NET Exception Handling Guide*).

## Main Content Chapters (Body)

You will include the following elements in each chapter:

- **Chapter introduction**
  - One to three sentences explaining the following:
    - Why this material is important to the document's subject
    - How it relates to any previous chapters
  - Include bulleted list or bulleted brief description of sections (and sometimes subsections) you will cover in that chapter. Each bullet will correspond to (and have similar or the same wording as) the h4-style section headings in the chapter. See the examples below from two different documents:

*Rebar QuickPort Web Application Deployment Guide*      *Chapter 2: Configure the Web Installer Project*

---

### Chapter 2: Configure the Web Installer Project

---

**Introduction**      You must **configure your web installer project** so that it contains all of the elements needed to enable your Rebar web application to run on another server. You will take four **main steps** in configuring the installer project:

- **Update project dependencies**
  - Configure installer project properties.
  - Configure web application folder properties.
  - Configure the output file folder name.

---

**Update Project Dependencies**      Follow these steps in Visual Studio to update your project dependencies:

1. In the **Solution Explorer** pane of Visual Studio, right-click the **installer project** (example: `SecurityAdmin_Installer`) and choose **Add ► Project Output**

*Rebar .NET Exception Handling Guide*      *Chapter 2: Overcoming .NET Exception Handling Challenges*

---

### Chapter 2: Overcoming .NET Exception Handling Challenges

---

**Introduction**      In Chapter 1, you saw the five requirements you must meet when coding the error trap approach. Now, you will look at the challenges faced when using in Microsoft's error trap mechanisms and how to overcome them using the **EDS.EIT.Common.Exception object** mechanism:

- **Getting the complete Call stack**
  - Avoiding Call stack information overload
  - Getting enough information to solve production issues
  - Working with COM and other environments
  - Deciding whether to use the Exception Management Application Block (EMAB) shareware that can be downloaded from Microsoft's web site

First, you will look at the **three Call stack mechanisms** in .NET.

---

**Call Stack Mechanisms**      Three native mechanisms are built into Microsoft's .NET for obtaining "Call stack" information:

- **StackTrace** property (built into all objects that inherit from `System.Exception`)
- **Tostring** (built into all objects that inherit from `System.Exception`)
- **StackTrace** property of the `System.Environment` class

All three Microsoft mechanisms exclude important information and can be

**Tables**

The following are various examples of if-then and other tables from different Rebar documents. **(You can copy these and use them as “templates” to be sure you have all subtle formatting included.)** Table formatting guidelines follow these examples:

Example 1—If-Then Table

1. Open **Add/Remove Programs**.

<b>If:</b>	<b>Then:</b>
You are running Windows <u>2000</u> ...	Click the <b>Start</b> button at the bottom left of your Windows screen to bring up the Start menu. Then choose in this order: <ul style="list-style-type: none"> <li>▪ <u>Settings</u></li> <li>▪ Control Panel</li> <li>▪ Add/Remove Programs</li> </ul>
You are running Windows <u>XP</u> ...	Click the <b>Start</b> button at the bottom left of your Windows screen to bring up the Start menu. Then choose in this order: <ul style="list-style-type: none"> <li>▪ Control Panel</li> <li>▪ Add/Remove Programs</li> </ul>

Example 2—If-Then Table

3. In the **Windows Components Wizard** window, find **Internet Information Services** in the Components list. Is the Internet Information Services (IIS) check box selected?

<b>If:</b>	<b>Then:</b>
IIS check box <u>is</u> checked...	Click the <b>Cancel</b> button and go on to <a href="#">Chapter 2: Install Visual Studio .NET</a> on page 18. IIS is already installed, and you are finished with this part of the process.
IIS check box is <b><u>NOT</u></b> checked...	<b>GO</b> to <a href="#">Step 4</a> below.

Example 3—If-Then Table

1. Have you added the web installer project to Visual SourceSafe yet?

<b>If:</b>	<b>Then:</b>
You <u>have</u> already added the <b>web installer project</b> to Visual SourceSafe...	<b>GO</b> on to <a href="#">Step 3</a> on page 19 below.
You have <b><u>NOT</u></b> yet added the <b>web installer project</b> to Visual SourceSafe...	<b>GO</b> to <a href="#">Step 2</a> below.



**Example 4—If-And-Then Table**

3. Look at the **Properties** dialog box that appears (example: eDogs Properties):

If:	And	Then:
Directory tab of the <b>Properties</b> dialog box does <b>NOT</b> contain a Create button...	<b>Application name</b> field contains <u>your project's application name</u> (example: eDogs)...	<b>GO</b> to <a href="#">Step 4</a> below.
Directory tab of the <b>Properties</b> dialog box does <b>NOT</b> contain a Create button...	<b>Application name</b> field is <u>blank</u> ...	<ul style="list-style-type: none"> <li>▪ Click the <b>Remove</b> button (a Create button will appear in its place).</li> <li>▪ Click the <b>Create</b> button to create the application.</li> <li>▪ <b>GO</b> to <a href="#">Step 4</a> below.</li> </ul>
Directory tab of the <b>Properties</b> dialog box <u>does</u> contain a Create button...		<ul style="list-style-type: none"> <li>▪ Click the <b>Create</b> button to create the application.</li> <li>▪ <b>GO</b> to <a href="#">Step 4</a> below.</li> </ul>

**Example 5—Time-to-Complete-Document Table**

Chapter	Approximate time (minutes) <b>IF</b> all preparation tasks are done beforehand
1 Create the Web Installer Project	10
2 Configure the Web Installer Project	15
3 Compile the Web Installer Project	05
4 Deploy the Web Application Project	15
<b>Total</b>	<b>45</b> <b>(approximately 1 hour)</b>

**Table Format:**

- Header (**table-head** style)—Arial 10 pt, bold, flush left (2pt above and below paragraph); 15 % black (gray) shading; initial capped (title case)
- Text (**table-text** style)—Arial 9 pt., flush left, sentence style capitalization
- Bulleted text (**table-bullet** style)—Arial 9 pt., bulleted, sentence style capitalization

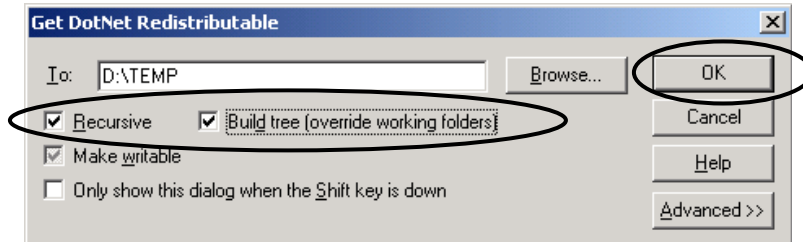
**Table Wording and Text Styles:**

- **NOT**—All caps, underlined, bold.
- **GO**—All caps, bold.
- **Differences** in If column rows—Underline the key difference in each If column option; (example: 2000 vs. XP; is vs. is **NOT**).
- **Key words**—Bold (as you would in step instructions in regular text).
- **If column** (and **And column** if one exists)—Follow text with an ellipsis (...).
- References (make these **hyperlinks** if they are the specific destination)
  - **Chapter** name to GO to—Underlined, initial capped (title case)
  - **Section** name to GO to—Underlined, quotation marks, initial capped (title case)
  - **Step** to go to—Underlined, initial capped (title case).
  - **Page number reference**—Add page number reference after “**GO** to...” instructions (for users who may be using a hard copy of the document). Be sure they are updatable fields so you do not have to maintain them if the document’s pagination changes, as in when a user prints on a different type of printer than yours or if the document is revised.

### Graphics and Screen Shots (all OH style):

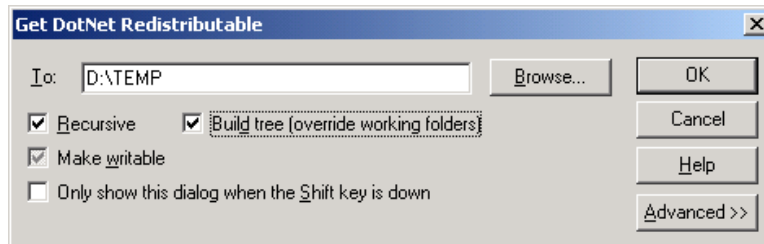
Use the following format when inserting screen shots or other graphics:

- When inserting a graphic, use Edit▶Paste Special... as “**picture**”—gives better quality, smaller file size.
- Use **OH style**. (can use **OH-large** for extra-wide graphics).
- Use a **circle** (black, 1.5 pt. drawing object) **to highlight** parts of window/box where user will take an action. Do **not** highlight simple “Next” or “OK” type buttons unless there are several choices.

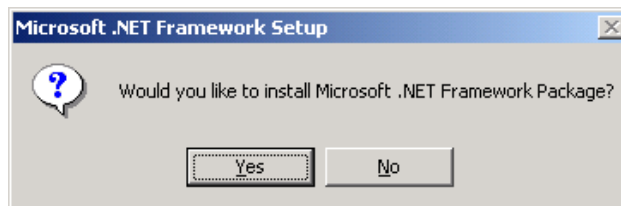


Use these guidelines to determine the **width** of the graphic so that most are the same width:

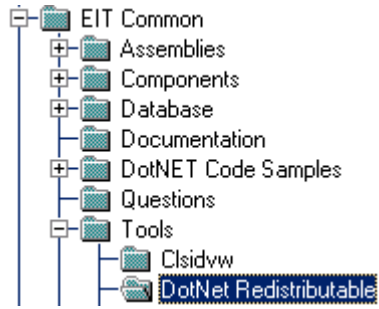
- Largest size—Can go all the way to the right margin
- Normal size—4”
- Smaller sizes—For screens/windows/boxes that do not need to be completely legible (if they only need to show the general look/layout of the screen for identification)—3.5”, 3.0”
- Extremely vertical windows, such as Solution Explorer—2.5”
- Extremely horizontal dialog boxes—4”



- Extreme horizontal dialog boxes (simple Yes/No, Okay/Cancel)—3.25”



- File tree—Leave its size as is (100%), since these are small.



**How to  
Organize  
Content**

One of the main reasons the format used in these documentation standards was chosen is because it helps you organize your content into chapters, sections, and subsections. Once you've put your draft content into this format, you can use the automatically-generated table of contents to see where you need to rearrange, combine, or divide material since the table of contents acts as an outline for the contents.

Try following these steps when organizing your content:

- Make a rough outline of your content in any format (using PC or handwriting notes on paper).
- Put your content into this standard Architecture Services format, even if you have some missing information.
- Update the table of contents.
  - **Right-click** the table of contents and choose **Update Field** from the popup menu that appears.
  - In the **Update Table of Contents** dialog box, choose **Update Entire Table** and click the **Okay** button.
- Look at the new table of contents; you can easily see the following:
  - If you have too much or too little information for a chapter, section, or subsection.
  - If same-level headings/topics are grouped with other similar topics.
  - If you are using consistent and parallel wording in your headings.

## Chapter 5: To Do Before Finalizing Document

---

**Introduction** This chapter lists the steps you should follow to prepare your documentation for reviews and release.

- Overall document cleanup and proofing
  - Preparation for release
- 

**Cleanup** You should do all of the following before sending documentation to a review (user review, conformance review) and before release:

- Update version # and date on title page.
- Update Amendment History page (don't forget version #s).
- Update document filename (change version # at end of filename).
- Update table of contents (update entire table, not just page numbers).
- Check table of contents for inconsistent formatting, bad chapter numbers, etc.
- Check document headers to be sure they match chapter numbers and chapter titles.
- Search for any colored fonts you or reviewers have used to make comments; use the Edit►Find function after you have looked visually.
- Remove any draft headers.
- Check for bad page breaks.
- Check all step numbers.
- Check all page number references in text.
- Be sure all text references to chapter, section, and step numbers are correct.
- Use Edit►Select All function and update all document fields (F9).
- Search for “ ’ “ to be sure there are no contractions (can also use Formal or Technical style in the Spelling and Grammar check tool and be sure contractions are selected as an item to check for).
- Last—Perform spelling and grammar check.

**Prep for  
Review or  
Release**

---

You should do all of the following before releasing the final document:

- Make document password-protected read-only. (In Tools>Options>Save tab, in File sharing options for ... fields, type a password in the “**Password to modify**” field. **Warning:** Be sure you remember or document this password, because otherwise you will not be able to edit the file again using the same file name.)
- Take any custom toolbars/buttons out of document (View>Toolbars... menu).
- Change document options (Tools>Options) to the following:
  - Update fields when printing (Print tab)
  - Embed TrueType fonts—characters in use only (Save tab)
  - Do not show formatting marks (View tab)
  - Turn off “check grammar as you type” and “check spelling as you type” so that squiggly underlines do not appear under questionable words/phrases. (Spelling and Grammar tab)
- Do not forget final spelling and grammar check!
- Last—Save document again one final time to make sure all of these options are what the user sees when he opens the document.

## Summary and Help

---

### Guide Summary

This guide has documented the **standards, formatting, and style guidelines** you should use when creating or editing Architecture Services documentation.

- Overall guidelines and references
  - Terms and usage
  - Fonts
  - Document format, elements, and organization
  - Tasks you must perform before finalizing a document
- 

### Support— Template and Examples

There are two ways to get support when using the Architecture Services documentation standards:

- **Arch\_Services\_DocStandards\_Template.dot** MS Word template. (This template contains some Rebar-specific content but can easily be modified.) See the [embedded template](#) in Chapter 1: Overall Guidelines and References on page 2.
  - **Examples of documentation using this format**—You can see finished documentation created in this format on line at <http://eads.eit.eds.com/EADS/development/rebar/doc.aspx>:
    - *Rebar ADO.NET Data Access Guide*
    - *Rebar QuickPort Web Application Deployment Guide*
    - *Rebar Config.xml Guide*
    - *Rebar .NET Exception Handling Guide*
    - *Rebar QuickPort Developer's Guide*
    - *Rebar QuickPort Installation Guide*
    - *Rebar Prerequisite Software Installation Guide*
- 

### Contact Us

The *Architecture Services Documentation Standards* is a living document that will evolve over time with additions and changes. Therefore, we welcome and need your suggestions for improving the guide. Please contact us at [EIT - DS Arch\\_Services MB](#) with your suggestions, corrections, or questions.