



application frameworkTM

version 5

installation guide



Installation Guide

**Zinc® Application Framework™
Version 5**

Zinc Software Incorporated
Pleasant Grove, Utah

NOTICE

This documentation is available in electronic and printed formats. If the electronic documentation is printable, a single copy may be printed for use by the Developer. Except for the foregoing, no part of this publication may be reproduced, translated, stored in a retrieval system, or transmitted, in any form or by any means, without the prior written permission of Zinc Software Incorporated (“Zinc”).

DISCLAIMER

While every precaution has been taken in the preparation of this manual, Zinc assumes no responsibility for errors or omissions. This publication and features described herein are subject to change without notice. ZINC MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREIN AND SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE.

TRADEMARKS

Zinc is a registered trademark and Zinc Application Framework, Zinc Designer and Zinc DataConnect are trademarks of Zinc Software Incorporated. All other trademarks and tradenames used herein are owned by their respective holders.

LICENSE AGREEMENTS

Zinc Application Framework is licensed subject to the terms and conditions of one of two separate license agreements found in this manual and in the “Getting Started” manual. The Personal Version license is provided to individuals developing non-commercial, non-distributable, personal-use-only applications. There is no license fee or royalty required for the Personal Version license. HOWEVER, TO EXERCISE RIGHTS BEYOND THE PERSONAL VERSION LICENSE, THE DEVELOPER MUST PURCHASE A PROFESSIONAL VERSION LICENSE FROM ZINC.

ACKNOWLEDGMENTS

The ChartFolio framework used by ZafChart is licensed software ©1994-97 DPC Technology Corporation. The XPM library used by ZafImage on Motif is licensed software ©1989-95 GROUPE BULL. The MetaWINDOW graphics primitives used by ZafDisplay on DOS is licensed software ©1988-96 Metagraphics, Inc.

This manual was generated March 18, 1999.

Copyright © 1990-1999 Zinc Software Incorporated.
All Rights Reserved.
Printed in the United States of America on recycled paper.

Contacting Zinc

Worldwide

Sales: info@zinc.com, sales@zinc.com
Technical Support: support@zinc.com
Training and Consulting: services@zinc.com
Web: <http://www.zinc.com/>
Ftp: <ftp://ftp.zinc.com/>

North America

Zinc Software Incorporated
405 South 100 East
Pleasant Grove, Utah 84062 USA
Tel: 1-801-785-8900
Sales: 1-800-638-8665
Support: 1-801-785-8998
Fax: 1-801-785-8996

Zinc Software Services, Inc.
42627 Garfield, Suite 214
Clinton Township, Michigan 48038 USA
Tel: 1-810-228-4900
Fax: 1-810-228-6633

Europe

Zinc Software (UK) Ltd.
106-108 Powis Street
London, SE18 6LU United Kingdom
Tel: +44 (0)181 855-9918
Fax: +44 (0)181 316-2211
Email: europe@zinc.com

Table of Contents

Introduction	5
Downloading Zinc 5	6
Licensing Zinc 5	12
Installing Zinc 5 from CD-ROM	16
Installing Zinc 5 from Downloaded Files	19
Configuring your Development Environment	23
Building Libraries	28
Building Example Programs	35
Learning Zinc Application Framework	37
Obtaining Technical Support	38
Personal Use License - Desktop Applications	39
Professional Use License - Desktop Applications	43
Professional Use License - Embedded Applications	47

Introduction

Thank you for selecting Zinc Application Framework version 5!

ZAF is the most advanced C++ class library ever developed for the creation of portable and international user interfaces. In addition to the class library, ZAF 5 includes Zinc Designer (a visual design tool), utility programs, examples, tutorials, and extensive electronic documentation.

Installing ZAF 5 is more than just copying files or running a simple installation program. The ZAF 5 installation process is designed to provide an important introduction to the Zinc development environment—including building class libraries and example programs from supplied source code. Although the installation process may seem detailed, developers will find the information gained during installation to be invaluable when using ZAF 5 for application development.

This installation guide provides step-by-step instructions for

- downloading ZAF 5, if not obtained on CD-ROM
- reviewing ZAF license agreements
- installing ZAF on your system
- configuring your development environment
- building ZAF libraries of your choosing
- building ZAF example programs
- obtaining technical support, if necessary

Some chapters in this “Installation Guide” are divided into sections corresponding to platforms currently supported by ZAF 5. Follow *only* those instructions that pertain to the platform(s) for which you are installing.

Free installation technical support is available to all users of ZAF 5. Please contact Zinc immediately if you have questions during this process. Complete technical support information can be found in the “Obtaining Technical Support” section of this guide.

Enjoy Zinc Application Framework version 5!

Downloading Zinc 5

Zinc Application Framework is available on CD-ROM, or it may be downloaded from many electronic forums including

- Zinc's [Web](#) site
- Zinc's [FTP](#) server
- many of Zinc's authorized resellers and partners
- other public forums

Before downloading any files, consult the "[Required Files](#)" section of this chapter. Instructions for downloading from Zinc's sites follow.

CD-ROM

Developers with CD-ROM distributions may skip this chapter.

All Zinc "Professional Version" customers automatically receive CD-ROM media which includes all platform modules. Members of Zinc's Customer Assurance Plan (CAP) receive regular upgrades on CD-ROM which also include all platform modules. However, a license key is required to install various professional version modules. For more information about different Zinc versions and software licensing, please refer to the "Installing Zinc" section of this manual, and the license agreements found at the end of this manual.

To purchase Zinc on CD-ROM, contact Zinc Software. Contact information can be found in the "Contacting Zinc" section of this manual.

Web

Zinc's home page on the world wide web may be reached via:

`http://www.zinc.com`

To download the Personal (Free) Version, select "Download Free Version" and follow the instructions. Additional software is available for downloading by those who "Register" their Personal Version (available online) and by those who purchase a "Professional Version" license (including those members of Zinc's "Customer Assurance Plan (CAP)") by selecting the download areas for these customers. Passwords are required for these areas of the web.

NOTE: The most current download instructions will *always* be found on the web itself.

FTP

The Zinc FTP server may be reached at:

`ftp.zinc.com`

Logins are accepted using any ftp client including WINFTP, FTP, or a web browser such as Netscape Navigator or Microsoft Internet Explorer. Anonymous browser logins usually take the following form

```
ftp://ftp.zinc.com
```

To download registered and professional versions of Zinc, or to access member-only software you will need to use a restricted login. Typical ftp syntax for popular browsers is

```
ftp://username:password@ftp.zinc.com
```

If requested, enter “anonymous” as your login name, and use your complete email address when prompted for a password. Once authenticated, move into the ZAF 5 distribution directory “/zaf.” Command-line users will enter

```
cd /zaf
```

Under the zaf directory are several subdirectories corresponding to free, registered, and professional versions of Zinc, among other directories. Change directories again into the appropriate directory. For example, command-line users might enter

```
cd /professional
```

Before downloading files, be sure to set “binary transfer” mode. Command-line users will enter

```
bin
```

Finally, download the necessary files. Command-line users will enter

```
get <filename>
```

(or for multiple files)

```
mget <filespec>
```

Required Files

Zinc Application Framework version 5 is a sophisticated product. As a portable development tool, much of its source code is shared between the various platforms supported by Zinc. This source code is referred to as “Shared Code.” Additional, non-portable code is also required to implement the portable functionality on each platform. This source code is referred to as “Platform Specific” code.

When downloading Zinc 5 (or an update), you must obtain both Shared Code and Platform-Specific Code. In addition, documentation and example pro-

grams are extremely helpful when using Zinc and are the most frequently updated portions of the product.

The following sections list the various archive file formats provided by Zinc Software. Multiple formats are provided to make file access as easy as possible for programmers on all platforms. Each file is provided in the format most likely to appeal to programmers using that format. For example, the “ZIP” format is the format most often used by developers on Windows, therefore Zinc supplies code for Windows platforms in this format.

Each table lists all files available in a given format. Files required for proper use of Zinc Application Framework are indicated by the word “*required.*” Other “*optional*” files may be downloaded as well.

If downloading ZAF for multiple platforms, you only need to download “Shared” files once. These may be used, without change, on all Zinc supported platforms.

To simplify downloads, we have included all required files into a few archives. The shared files are found in `shared.zip` and `shared.tar.Z`, and platform-specific files are found in `windows.zip` and `motif.<os>.tar.Z`. For example, to download required files for Zinc on Microsoft Windows, you need only download `shared.zip` and `windows.zip`. These files’ contents are explained below.

ZIP Archive Format
(PC Platforms)

Zinc’s PKZIP archives contain Shared code (portable to all platforms) and platform-specific code for Microsoft Windows as well as examples and tutorials, electronic documentation, etc.

File	Contents
<code>zafinc.zip</code>	Shared (portable) header files. <i>Required.</i>
<code>zafsrc.zip</code>	Shared (portable) library source code files. <i>Required.</i>
<code>zafuni.zip</code>	Shared (portable) Unicode files (non-personal versions). <i>Required for Unicode.</i>
<code>???reg.zip</code>	Notebook class source code files (non-personal versions). <i>Required for ZafNotebook.</i> Replace “ <code>???</code> ” with either “ <code>win</code> ” or “ <code>mtf</code> ”. These files are not required for embedded versions of Zinc.
<code>???pro.zip</code>	Table, splitter, printer and notebook class source code files (professional version). <i>Required for these objects.</i> Replace “ <code>???</code> ” with either “ <code>win</code> ” or “ <code>mtf</code> ”. These files are not required for embedded versions of Zinc.
<code>zafdoc.zip</code>	Online documentation (professional version). <i>Strongly recommended for professional version.</i>

File	Contents
zafdocn.zip	Online documentation (non-printable, non-indexed). <i>Strongly recommended for non-professional versions.</i>
zafread.zip	Readme text files. <i>Required.</i>
wininst.zip	Installation files for Microsoft Windows. <i>Required for Windows.</i>
winsrc.zip	Windows library source files. <i>Required for Windows.</i>
winbin.zip	Windows binary files: Zinc Designer, utilities, data, etc. <i>Required for Windows.</i>
indsrc.zip	Shared library source files for <i>independent</i> versions of Zinc. <i>Required for DOS, pSOS+, VxWorks, and custom OS ports.</i>
dosinst.zip	Installation files for DOS. <i>Required for DOS.</i>
dossrc.zip	DOS library source files. <i>Required for DOS.</i>
dosbin.zip	DOS binary files: Zinc Designer, utilities, data, etc. <i>Required for DOS.</i>
psssrc.zip	pSOS+ library source files. <i>Required for pSOS+.</i>
vxwsrsrc.zip	VxWorks library source files. <i>Required for VxWorks.</i>
acrobtt?.exe	Adobe Acrobat reader with search capability for Windows. <i>Required to view electronic documentation.</i> May be obtained from other sources including Adobe at http://www.adobe.com
pkunzip.exe	Uncompresses files with zip extension. <i>Required to uncompress files.</i> May be obtained from other sources.
zafexm.zip	Example programs and tutorials. <i>Strongly recommended.</i>
zafconv.zip	ZAF 4 to ZAF 5 data file conversion utilities, source code, and example conversions. <i>Optional.</i>
zafattr.zip	Library test suite source code (attribute tests). <i>Optional.</i>
cf_zaf.zip	Third-party ChartFolio charting library source files. <i>Optional, but required for use of the ZafChart object.</i>
shared.zip	cf_zaf.zip, zafexm.zip, zafinc.zip, zafread.zip, zafsrc.zip
windows.zip	winsrc.zip, winbin.zip, wininst.zip
dos.zip	dossrc.zip, indsrc.zip, dosbin.zip, dosinst.zip
psos.zip	psssrc.zip, indsrc.zip, wininst.zip
vxworks.zip	vxwsrsrc.zip, indsrc.zip, wininst.zip

tar.Z Archive Format
(Unix Platforms)

Zinc's tar archives contain Shared code (portable to all platforms) and platform-specific code for X/Motif as well as examples and tutorials, electronic documentation, etc.

File	Description
zafinc.tar.Z	Shared (portable) header files. Required.
zafsrc.tar.Z	Shared (portable) library source code files. Required.
zafuni.tar.Z	Shared (portable) Unicode files (non-personal versions). Required for Unicode.
zafreg.tar.Z	Notebook class source code files (non-personal versions). Required for ZafNotebook.
zafpro.tar.Z	Table, splitter, printer and notebook class source code files (professional version). Required for these objects.
zafdoc.tar.Z	Online documentation (professional version). Required for professional version.
zafdocn.tar.Z	Online documentation (non-printable, non-indexed). Required for non-professional versions.
zafread.tar.Z	Readme text files. Required.
mtfinst.<os>.tar.Z	Installation files for Motif. Replace "<os>" with an appropriate operating system identifier. Required for Motif.
mtfsrc.tar.Z	Motif library source files. Required for Motif.
mtfbin.tar.Z	Motif binary data files. Required for Motif.
mtfbin.<os>.tar.Z	Motif Zinc Designer executable. Replace "<os>" with an appropriate operating system identifier. Required for Motif.
[acrobat reader]	Adobe Acrobat reader with search capability for your platform. Required to view electronic documentation. May be obtained from other sources including Adobe at http://www.adobe.com
zafexm.tar.Z	Example programs and tutorials. <i>Strongly recommended.</i>
zafconv.tar.Z	ZAF 4 to ZAF 5 data file conversion utilities, source code, and example conversions. <i>Optional.</i>
zafattr.tar.Z	Library test suite source code (attribute tests). <i>Optional.</i>
cf_zaf.tar.Z	Third-party ChartFolio charting library source files. <i>Optional, but required for use of the ZafChart object.</i>
xpm.tar.Z	Third-party XPM library source files for Motif. <i>Optional, but required for use of the ZafImage object.</i>

File	Description
shared.tar	cf_zaf.tar.Z, zafexm.tar.Z, zafinc.tar.Z, zafread.tar.Z, zafsrc.tar.Z
motif.<os>.tar	mtfbn.tar.Z, mtfbin.<os>.tar.Z, mtfinst.<os>.tar.Z, mtfsrc.tar.Z, xpm.tar.Z
psos.tar	indsrc.tar.Z, psssrc.tar.Z. <i>mtfinst.<os>.tar.Z must also be downloaded.</i>
vxworks.tar	indsrc.tar.Z, vxwsrsrc.tar.Z. <i>mtfinst.<os>.tar.Z must also be downloaded.</i>

Licensing Zinc 5

Zinc Application Framework version 5 is available under the terms of three different license agreements. Before actually installing Zinc 5, an understanding of basic licensing terms is necessary. *During the upcoming installation process you will be asked if you have read and accepted the terms and conditions of the appropriate license agreement.*

Notice

THIS SECTION OF THE “INSTALLATION GUIDE” IS INTENDED TO PROVIDE A PLAIN ENGLISH DESCRIPTION OF MAJOR LICENSE AGREEMENT PROVISIONS. IT IS NOT COMPLETE AND NEITHER SUBSTITUTES FOR, NOR SUPERCEDES THE LICENSE AGREEMENTS THEMSELVES. YOU ARE REQUIRED TO READ THE APPROPRIATE LICENSE AGREEMENT AND CONSENT TO ITS TERMS AND CONDITIONS DURING THE INSTALLATION OF ZINC 5. COMPLETE LICENSE AGREEMENTS MAY BE FOUND AT THE END OF THIS MANUAL.

Desktop OS License Terms

Zinc’s Professional Use License for Desktop Applications includes terms and conditions which are common among professional desktop development tools. A few provisions worth mentioning include

- Per seat, per platform licensing
Zinc 5 is licensed on a per-developer, per-platform basis (i.e., a license to the Shared Code is required for each individual developer as well as a license to the Platform Module for each platform used by that developer). Multiple developers require multiple Shared Code licenses, each with the appropriate Platform Module license(s) corresponding to the platform(s) used by that developer.
- Zinc copyright notice requirement
All applications developed using Zinc Application Framework must display a valid Zinc copyright notice (see below).
- Non-compete requirement
Zinc 5 may not be used to develop an application that is competitive with, or which may be used in lieu of Zinc 5.

Embedded OS License Terms

Zinc’s Professional Use License for Embedded Applications includes provisions which are appropriate and customary for the embedded marketplace but which differ greatly from the desktop world. These provisions include

- **Per-project licensing**
The project license includes target software which can be used by any number of developers on the specified project and host software which is licensed on a per developer basis (similar to desktop licensing).
- **Target license fees (royalties)**
Target runtime license fees must be paid for each embedded application (or device) that includes Zinc code.
- **Zinc copyright notice requirement**
All applications developed using Zinc Application Framework must display a valid Zinc copyright notice (see below).
- **Non-compete requirement**
Zinc 5 may not be used to develop an application that is competitive with, or which may be used in lieu of Zinc 5.

Personal (Free) Version

Zinc's Personal Use License for Desktop Applications allows individuals to

- use the Software to develop non-commercial, non-distributable, personal-use-only applications (i.e., the only end-user of the applications you build with the Personal Version software is you).
- distribute complete and unmodified copies of the Personal Version software as distributed by Zinc (i.e., only those Personal Version distribution files as found on Zinc's web server <http://www.zinc.com/>).
- distribute source code to applications developed using the Personal Version software (executables may not be distributed).

ANY OTHER USE NOT LISTED ABOVE REQUIRES THE PURCHASE OF A PROFESSIONAL USE LICENSE FOR EITHER DESKTOP OR EMBEDDED APPLICATIONS.

Professional Version

In addition to Personal Version benefits, the Professional Version license agreements allow individuals and entities to use the Zinc Application Framework Professional Version software designated in the License Certificate to

- develop applications which can be distributed in executable or source code form (only embedded applications require a run-time fee);
- distribute some Zinc binary files with a Zinc-based application (e.g., a Zinc DLL or shared library);
- print reference documentation for personal use;
- gain access to Zinc professional technical support and the Customer Assurance Plan.

EACH PROFESSIONAL VERSION DEVELOPER MUST BE PROPERLY LICENSED FOR ALL PLATFORM(S) USED BY THAT DEVELOPER.

Registered Software

As a benefit of registering their software, “Personal Version” developers gain access to additional Zinc objects and functionality. Likewise, “Professional Version” developers have access to still more objects and functionality.

Restricted-access software components may not be re-distributed. All other Zinc software (i.e., software obtained without passwords or other restrictions) may be freely re-distributed *in its original form* (i.e., the original archive file(s)).

Shipping Your Application

Be sure to include the following run-time files (if applicable) when you distribute your finished application:

- *.znc files (generated by Zinc Designer) required by your application.
- i18n.znc (ISO only or Unicode/ISO version) required by internationalized applications.
- Dynamically-linked library file (e.g., zafw32.dll or libZafMtf.so) required by applications which run-time load the Zinc library.
- MetaWINDOW font files (*.fnt) and drivers (*.drp) required by DOS graphics applications which use the MetaWINDOW display class. Or UGL font file (*.fnt) required by embedded applications which use the UGL display class.

YOU MAY NOT INCORPORATE INTO YOUR APPLICATION OR DISTRIBUTE AS PART OF YOUR APPLICATION ANY PORTION OF ZINC DESIGNER WITHOUT THE EXPRESS WRITTEN PERMISSION OF ZINC.

Zinc Copyright Notice

You must include an appropriate Zinc copyright notice, in accordance with guidelines published by Zinc, on all copies of your application(s). The most recent guidelines can be found on Zinc’s web site (<http://www.zinc.com/>). As of this writing, guidelines for displaying a valid Zinc copyright notice are as follows:

- Notice: “Portions Copyright © 1990-99 Zinc Software Incorporated (www.zinc.com)--All Rights Reserved.”
- Location: Below or to the side of each occurrence of your copyright notice (e.g., program startup or splash screen, help/about window, program media, program documentation, etc.).

- Size/visibility: Can be less conspicuous than your own notice but must be reasonably sized and readable.

Installing Zinc 5 from CD-ROM

Zinc Application Framework is available on CD-ROM or via electronic download. Installation methods differ. Please choose the one appropriate for you.

SEVERAL FILES NECESSARY FOR USING THE ZINC LIBRARIES ARE ENCRYPTED. YOU MUST RUN THE INSTALLATION FILES DESCRIBED BELOW TO BE ABLE TO USE THE ZINC LIBRARIES.

Downloaded Files

If installing from downloaded files, please consult the chapter “Installing Zinc 5 from Downloaded Files.”

CD-ROM

To install from CD-ROM you must first make the CD accessible on your system. When mounting the CD on a Unix system the CD must be mounted as an ISO-9660 file system or the system will be unable to read the Zinc CD. Please consult your operating system documentation or system administrator for assistance. Zinc cannot provide technical support for the attachment of CD-ROM devices.

Run Install

1. Ensure that you have full access privileges to the location you wish to use for Zinc Application Framework version 5.
You must be able to read, write, modify, and delete in the directory you will use for Zinc 5. Note: on Unix systems, the installation program will also use the “/tmp” directory for intermediate files.
2. Change your default workspace to be the CD-ROM.
 - On Windows, open a Win95/NT Explorer or Win3x File Manager window and select the CD-ROM. If running Windows 95 or NT you may also select the CD-ROM using a DOS prompt (see below). *Under Windows 95/98 or NT the autorun may start the installation program upon inserting the installation CD into the CD-ROM drive. If this happens, press “Next >” on the introduction screen to advance to the licensing window, then proceed with step 4.*
 - On Unix, or at a Windows/DOS prompt, use “cd” to select this directory. For example, “cd /mnt/cdrom”.
 - On MS-DOS, make the CD-ROM drive current.
3. Run the installation program.
 - On Windows, select and run the “setup” (Windows NT, 98, 95) or “setup16” (Windows 3.x) graphical installation programs. If running from a Windows/DOS prompt, type “setup” or “setup16”.

- On Unix, type “`setupmtf`” to run the install script. This script will determine the Unix system and call the correct install program.
- On MS-DOS, type “`setupdos`” to run the graphical installation program. After the introduction screen is presented, press “Next >” to advance to the licensing window.

4. Review and accept relevant Zinc license agreements.

Some or all of the license agreements will be relevant for each individual installation. Review the synopsis at the start of each agreement to determine which agreements apply to you.

Press “Accept” to indicate your agreement with the terms of the license agreement(s) and continue the installation, or press “Decline” to terminate the installation.

5. (optional) Provide installation keys.

Some Zinc software components require a customer ID and installation key in order to install and decrypt them. If you are installing any software other than a “Personal Version” of Zinc, you should enter this information. Your customer ID and installation key may be found on your Zinc invoice, license certificate, or installation addendum. After entering this information, press “Next >” to proceed.

6. Select the directory into which Zinc 5 should be installed.

This directory will be the “root” of the Zinc installation and will contain other Zinc subdirectories. You should note this directory as you will need the information later in the installation process to configure your development environment.

Software Components

7. Select the Zinc software components you wish to install.

The software components available for installation will depend on the installation key previously entered. If the component you need is disabled, press “< Previous” to return to the installation key window and re-enter the customer ID and installation key.

Some software components are really component sets comprised of multiple sub-components. To review and select subcomponents, press “Details.” A short description of each component and the selection status of subcomponents, if any, is shown in the box below the component list.

Select “Install” to proceed with the installation.

8. Review selected components. The editor window displays a list of all software components you have selected for installation. If this information is correct, press “Next >” to install the software, otherwise press “< Previous” to return to the components window and make changes.

9. During the actual automated installation you may press “Cancel” to terminate the process. Note that this will leave the installation in an undetermined state. Zinc will not “clean up” after a terminated installation so you must do this manually.
10. Review installation messages, including errors if any, in the “Finished” window.
11. (Unix only) After the file copying and decrypting is complete, you will be asked to answer several questions required to properly configure Zinc make files for your Unix development environment. If needed, this portion of the installation may be repeated separately later to change the configuration. Use “mconfig” for this purpose.
12. (All except Unix) Configure your development environment as discussed in the section titled, “Configuring your Development Environment.”

Configure development
environment

File Locations

You may want to familiarize yourself with the installed directory structure before continuing. The Zinc tree looks like this:

```
zaf (root of ZAF tree)
  attrtest (ZAF attribute test programs source)
  bin (PC designers)
    aix4 (AIX 4 designer)
    hpux9 (HP-UX 9 designer)
    hpux10 (HP-UX 10 designer)
    irix5 (Irix 5 designer)
    linux2030 (Linux 2 with GNU C++ 2.7.2.1, lib5c - designer)
    linux2031 (Linux 2 with GNU C++ 2.7.2.3, lib6c - designer)
    osf1 (Digital Unix designer)
    sco (SCO designer)
    sun4 (SunOS 4 designer)
    sun5 (Solaris 2 designer)
  cf_zaf (ChartFolio library source)
  doc (ZAF electronic documentation)
  example (ZAF example programs)
  include (ZAF library headers)
  readme (ZAF informational text files)
  source (ZAF library source)
    intl (international data file source and library source)
    unicode (Unicode support source and i18n.znc data file)
  xpm (Motif only--3rd party XPM library source)
  ugl (Universal Graphics Library--optional, “independent”
    platforms only)
  util (Zinc utilities)
    extract (text extraction utility for .znc data files)
    zncmerge (merge utility for .znc data files)
```

Installing Zinc 5 from Downloaded Files

Zinc Application Framework is available on CD-ROM or via electronic download. Installation methods differ. Please choose the one appropriate for you.

SEVERAL FILES NECESSARY FOR USING THE ZINC LIBRARIES ARE ENCRYPTED. YOU MUST RUN THE INSTALLATION FILES DESCRIBED BELOW TO BE ABLE TO USE THE ZINC LIBRARIES.

The .zip files are for PC installations, and the .tar.Z files are for Unix installations.

CD-ROM

If installing from CD-ROM, please refer to the chapter “Installing Zinc 5 from CD-ROM.”

ZIP files

1. Uncompress all files.

After downloading the files from any of our electronic sites, place them together in a directory chosen to be the root Zinc directory. Zinc 5 uses several subdirectories that are best organized underneath a Zinc parent directory. For example, a good choice for an installation directory might be “zaf5”.

If using WinZip to uncompress the downloaded ZIP files, open and extract (uncompress) each required .zip file, using the ZAF root directory as the target directory. Some .zip files contain a set of other .zip files. For example, windows.zip contains winbin.zip, wininst.zip, and winsrc.zip. For these container .zip files, simply extract them, then extract (uncompress) the resulting .zip files.

If using pkunzip, place the files in the chosen installation directory (ZAF root), make it the current directory, then use 'pkunzip -d *.zip' to unzip each file. This will create the entire Zinc tree and place the uncompressed files in their proper locations.

If you have downloaded the registered or professional version, be sure to uncompress ???reg.zip and/or ???pro.zip last, since they replace existing files in the tree. (“???” indicates the three letter platform abbreviation.)

2. Read and accept the license agreement.

Read and accept the appropriate license agreement found in this manual (and found in the license.txt file in the README directory). The license.txt file may be opened and viewed with any text editor.

3. Run the wininstall (MS Windows) or install (MS-DOS) batch file.

When run, these batch files prompt you to confirm that you have read and accepted the license agreement. To agree, choose “y”; otherwise choose “n” to terminate the installation. Upon acceptance, the batch files decrypt certain files necessary to use Zinc. If ZIP files for both MS Windows and MS-DOS have been downloaded, only one batch file need be run.

NOTE: If you don't specify “y” in the wininstall script at this time, you will be unable to build the Zinc libraries since some files will remain encrypted.

tar.Z files

1. Uncompress all files.

After downloading the files from any of our electronic sites, they should be placed in a directory chosen to be the root Zinc directory. Zinc 5 uses several subdirectories that are best organized underneath a Zinc parent directory. For example, an installation directory might be “/users/myaccount/zaf5”.

Uncompress the .Z files by using the Unix uncompress utility for each file. For example:

```
uncompress zafsrc.tar.Z
```

After the files have been uncompressed, unarchive each of them to create the Zinc tree.

If you have downloaded the registered or professional version, be sure to unarchive zafreg.tar and/or zafpro.tar last, since they replace existing files in the tree.

```
tar -xvf zafsrc.tar
```

The complete directory tree will be created and the Zinc files will be placed in the appropriate locations.

2. Read and accept the license agreement.

Read and accept the appropriate license agreement found in this manual (and in the license.txt file in the readme directory). The license.txt file may be opened and viewed with vi or any other text editor.

3. (Unix only) Run the setupmtf script.

- When run, the `setupmtf` script prompts you to confirm that you have read and accepted the appropriate license agreement. To agree, choose “y”; otherwise choose “n” to terminate the installation.

NOTE: If you don't specify “y” in the `setupmtf` script at this time, you will be unable to build the Zinc libraries since some files will remain encrypted.

- You are next prompted to select the system to install for. A numbered list of systems is presented. Type the number according to the system you are installing for. For example, IBM AIX is “4”.
- Next, you are prompted for the compiler for which the make files will be configured. A numbered list of compilers is presented. For most systems, this will be either the native compiler, or the GNU compiler. Type the number according to the compiler you will be using. For example, the IBM AIX native compiler is “1.”
- If the system and compiler support shared libraries, you will next be prompted to choose between archived and shared libraries; otherwise make files will be configured to build and use archived libraries.
- After making the previous selections, the current default values for the chosen system and compiler are displayed. These values will be typical for the selections that were made. Before modifying them, consult with the system administrator to be sure of the appropriate values. The installation script next modifies the source file extensions, examples, and make files to reflect the choices previously selected. Finally, `setupmtf` decrypts certain files necessary to use Zinc.

File Locations

You may want to familiarize yourself with the installed directory structure before continuing. The Zinc tree should look like this:

```
zaf (root of ZAF tree)
  attrtest (ZAF attribute test programs source)
  bin (PC designers)
    aix4 (AIX 4 designer)
    hpux9 (HP-UX 9 designer)
    hpux10 (HP-UX 10 designer)
    irix5 (Irix 5 designer)
    linux2030 (Linux 2 with GNU C++ 2.7.2.1, lib5c - designer)
    linux2031 (Linux 2 with GNU C++ 2.7.2.3, lib6c - designer)
    osf1 (OSF/1 designer)
    sco (SCO designer)
    sun4 (SunOS 4 designer)
    sun5 (Solaris 2 designer)
  cf_zaf (ChartFolio library source)
  convert (ZAF 4 to 5 conversion utilities and examples)
  doc (ZAF electronic documentation)
  example (ZAF example programs)
  include (ZAF library headers)
```

```
readme (ZAF informational text files)
source (ZAF library source)
  intl (international data file source and library source)
  unicode (Unicode support source and i18n.znc data file)
xpm (Motif only--3rd party XPM library source)
ugl (Universal Graphics Library--optional, "independent"
platforms only)
util (Zinc utilities)
  extract (text extraction utility for .znc data files)
  zncmerge (merge utility for .znc data files)
```

Configuring your Development Environment

Before building Zinc libraries or applications the development environment must be configured. Configuration consists of setting environment variables with information about your compiler. This chapter is divided into sections for PC compilers, Unix compilers, Integrated Systems' pSOS+, and Wind River's Tornado. Refer to those sections applicable to your installation.

PC Compilers

Zinc provides a custom make utility called `zmake` (found in the BIN directory). Like most compiler-supplied make utilities, `zmake` relies on environment variables to find other necessary components such as compilers and linkers. Normally these environment variables should be set in startup files so they will always be available. **These environment variables must all be set before building any Zinc libraries or applications. Do not use spaces when setting these environment variables.**

Typically, Zinc is installed to a directory of its own. This directory then contains an organized set of subdirectories containing the Zinc 5 files. By default, this "Zinc root" directory is called "ZAF" or "ZAF???". In the example code that follows, "ZAF" is assumed to be the root Zinc directory. Replace this name if you installed to a different directory.

PATH The "PATH" environment variable lists the directories where the operating system looks for external programs requested from the command line. PATH must be modified to include the "BIN" directory where Zinc Designer, `zmake`, and other Zinc utilities are stored. For example:

```
set PATH=%PATH%;C:\ZAF\BIN
```

ZINC_COMPILER The "ZINC_COMPILER" environment variables tells `zmake` which vendor's compiler, linker, and librarian to use. Valid choices include BORLAND, MICROSOFT, or WATCOM. For example:

```
set ZINC_COMPILER=BORLAND
```

ZAF_PATH "ZAF_PATH" defines where Zinc's BIN directory is located. This is needed by Zinc Designer to find data files it depends on.

```
set ZAF_PATH=C:\ZAF\BIN
```

ZAF_ROOT "ZAF_ROOT" specifies the location of Zinc's root directory. This variable is used by the `zmake` utility to find header and source files when building a Zinc library or application.


```
set ZAF_ROOT=C:\ZAF
```

INCLUDE

“INCLUDE” is an environment variable commonly used by compilers other than Borland. It is used to locate header files. Be sure to include the compiler's include directories as well. Although Borland does not use environment variables, the zmake utility uses them, so INCLUDE should be set for all compilers including Borland.

```
set INCLUDE=C:\BC5\INCLUDE;C:\ZAF\INCLUDE
```

If you will be using the optional, supplied ChartFolio libraries (required to use the ZafChart object) you should also add

```
;C:\ZAF\CF_ZAF\INCLUDE
```

LIB

The “LIB” variable is also common for compilers other than Borland. It is used to locate library files. Be sure to include the compiler's library directories as well. Although Borland does not use environment variables, zmake uses them, so LIB should be set for all compilers including Borland.

```
set LIB=C:\BC5\LIB;C:\ZAF\LIB\BORLAND
```

To provide support for DOS filenames, Zinc's filenames and directories adhere to the 8.3 naming format. Because of this, the Zinc directory for compiled Microsoft libraries is named LIB\MICROSFT instead of LIB\MICROSOFT. Therefore when using a Microsoft compiler, set the LIB variable as follows:

```
set LIB=C:\MVC5\LIB;C:\ZAF\LIB\MICROSFT
```

CF_ZAF

“CF_ZAF” specifies the location of the root directory of ChartFolio for Zinc. This variable is used by the zmake utility to find header and source files when building a Zinc library or application that uses ChartFolio for Zinc.

```
set CF_ZAF=C:\ZAF\CF_ZAF
```

PHARLAP

“PHARLAP” specifies the location of the root directory of Phar Lap DOS Extender. This variable is used by the zmake utility to find Phar Lap runtime files when building a Zinc DOS application that uses Phar Lap.

```
set PHARLAP=C:\RUN286
```

**Unix
Compilers**

Configuring Zinc for Unix environments is handled automatically by the “mconfig” script. If you installed Zinc using the graphical installation program, your environment has already been configured.

Unlike PC platforms, Zinc for Unix platforms uses native make files unique to each compiler. To reduce maintenance issues, Zinc provides generic make files called “posix.mak”. mconfig translates these posix.mak files to “Make-

file”s for each compiler. You may run `mconfig` at any time to reconfigure make files for a different compiler or build option.

Manually setting environment variables is not generally necessary on Unix, but you may wish to set a few as described below.

ZAF_PATH	“ZAF_PATH” should be defined to allow Zinc Designer to be run from locations other than “zaf/bin”. Its definition is the same on both PC and Unix platforms.
CF_ZAF	“CF_ZAF” specifies the location of the root directory of ChartFolio for Zinc. Its definition is the same on both PC and Unix platforms.

Integrated Systems pRISM+ and pSOS+

Using Zinc in Integrated System’s pRISM+ development environment requires additional configuration as indicated below. Zinc directly supports the CadUL Organon toolset for x86 that ships with pRISM+ version 1.2.3. Zinc also works with earlier versions of pRISM+ and pSOS+ (although the versions of the `sys_conf.h` and `drv_conf.c` files used by the source/test and example/hello programs will need to be modified in this case.) However, you will need the newer version of the CadUL tools to build C++ applications. Other compilers should also work, although makefile modifications will be necessary.

Zinc has not yet been tested in a Sniff project in the pRISM+ IDE.

Verify pRISM+ installation

1. When pRISM+ 1.2.3 is installed it configures a Korn shell file that is automatically used to configure the development environment used in the pRISM+ DOS prompt shell. A bug in the Korn shell script may require you to set up a few environment variables since the CadUL tools may interpret environment variables including backslashes “\” as containing escape characters. A simple batch file can be used to work around this problem. For example:

```
set PSS_BSP = %PSS_ROOT%/bsps/pc
```

Zinc makefiles use the makefile include structure that ships with pRISM+. By default, the compile options set in `$(PSS_ROOT)/configs/std/configxx.mk` include debug information (`-VXDB`). For most developers, this options should be turned off before building the Zinc libraries. Debug information increases the size of the libraries by 15-20 times and substantially increases the time required to link applications. To eliminate this distraction, remove the “-VXDB” option from the following lines in `figxx.mk`:

```
COPTS =
CXXOPTS =
```

When creating pSOS+ libraries containing C++ objects the `-idp` library option must be used to avoid “duplicate public symbol” errors. Zinc uses this option by default in its makefiles.

Configure pSOS+

2. Configure pSOS+.

Include pREPC in your application by adding it to `sys_conf.h`.

If you will be using Zinc’s persistence support (recommended) you will also need to configure your `sys_conf.h` file to include a disk of some sort, as well as the pHILE disk file system.

Zinc environment variables

3. Configure required Zinc environment variables (some may not be required for Unix hosts.) For example:

```
set ZINC_COMPILER = cadul
set ZAF_ROOT = c:/zaf
set ZAF_PATH = c:/zaf/bin
set PATH = c:/zaf/bin
set CF_ZAF = c:/zaf/cf_zaf
```

UGL environment variables

4. Configure UGL environment variables. For example:

```
set UGL_ROOT = c:/zaf/ugl
set UGL_OS_HEADER = uosPsos.h
set UGL_ARCH_HEADER = uarX86.h
```

Wind River Tornado and VxWorks

Using Zinc in Wind River’s Tornado development environment requires additional configuration as indicated below. Zinc uses the Tornado GNU compiler.

Configure VxWorks

1. Configure VxWorks using WindConfig in the Tornado IDE, or by directly modifying `%WIND_BASE%/target/config/[target]/config.h`.

Add support for C++ by adding the line “`#define INCLUDE_CPLUS`”.

Add support for a disk file system if you intend to use Zinc’s built-in persistence support (recommended). To include support for POSIX file system access, add the line “`#define INCLUDE_FTP_SERVER`”. Also include a disk component in your configuration.

Be sure to include any other components that require BSP support as well.

Build bootrom and VxWorks

2. Build your bootrom and VxWorks system image as described in the VxWorks Programmer’s Guide.

Run torvars

3. Run `torvars`.

When you installed Tornado on your host, it created a batch file or script called `torvars`. Before you can build Zinc libraries or applications from the command line, you will need to run this batch file to configure your

environment so that makefiles can locate required components including the compiler, header files, etc.

Zinc environment
variables

4. Configure Zinc environment variables.

```
set ZINC_COMPILER = gnu
set ZAF_ROOT = c:\zaf
set ZAF_PATH = c:\zaf\bin
set PATH = c:\zaf\bin
set CPU = I80486      // use any valid VxWorks CPU identifier
```

Refer to the environment variable descriptions in the above sections for more details.

UGL environment
variables

5. Configure UGL (Universal Graphics Library) environment variables.

```
set UGL_ROOT = c:\zaf\ugl
set UGL_OS_HEADER = uosVxw.h
set UGL_ARCH_HEADER = uarX86.h
```

These values must be defined at compile time when building your application. Zinc defines them in the UGL library make files. “uarx86” specifies the processor architecture for UGL. Analogous values may be used depending on your target architecture. See the UGL documentation or source for details.

UGL is not the only low-level graphics library that can be used with Zinc applications for VxWorks. Metagraphics MetaWINDOW is also supported (for x86 platforms only). For MetaWINDOW, you should set the `METAPATH` environment variable to the directory where runtime files (such as fonts) can be found. If you will be using another graphics library, you should configure environment variables for that library.

Apply required patch

6. VxWorks for x86 version 5.3.1 has a critical bug in the DOS filesystem code. This must be patched for Zinc applications to properly access files. The required patch file is “dosfslib.obj” and may be found on updated Tornado CDROMs, or from Wind River technical support. The file may also be found on the Zinc ftp site at <ftp.zinc.com/zaf/contrib/zinc/fixes/vxwdosfslib.zip>.

The zip file includes patch instructions. Rebuild the VxWorks BSP after applying the patch.

Building Libraries

Zinc Application Framework Version 5 supports many different operating systems, compilers, library types, etc. In fact, the number of combinations is so large that Zinc cannot realistically build all possible combinations and ship them in a single release.

For example, on Microsoft Windows alone Zinc supports four versions of the operating system (Windows NT, 95, 98 and 3.x), three build options (16 bit static libraries, 32 bit static libraries, 32 bit DLLs), three compilers (Microsoft, Borland, and Watcom), three character sets (local code page, ISO-8859-1, and Unicode), and various other build options including debug or non-debug libraries. Clearly, any attempt to build a full set of libraries would rapidly exceed reasonable time and storage resources.

For this reason, and to ensure that users are comfortable building Zinc libraries, Zinc requires developers to build their own development libraries. The following instructions outline this process. Note: Building Zinc 5 libraries may require anywhere from 2 minutes to over one hour depending on the hardware and compiler used.

This chapter is divided into several sections:

- Modularization (all developers should consult this section.)
- PC Compilers
- Unix Compilers
- pRISM+ and pSOS+ (pSOS developers only)
- Tornado and VxWorks (VxWorks developers only)
- Third-party libraries

Consult only those sections that apply to your development environment(s).

Modularization

To allow greater control over Zinc's footprint, Zinc libraries may be built with various library components removed. All components are included in the Zinc libraries by default. Removing components will also remove functionality and therefore cause the library to work differently than documented.

Zinc modules are defined in `zinc/include/z_env.hpp`. Edit this file and comment or uncomment the following macro definitions to achieve the desired level of modularization and functionality.

Note: Module sizes are approximate and may vary significantly depending on operating system and compiler.

Macro	Module functionality
USE_ZAF_CODE_SET_DATA	Allow Zinc to map characters from a portable internal format (ISO-8859-1 or Unicode) to and from the local code page representation on each OS. (1K)
USE_ZAF_ERROR_SYSTEM	Allow Zinc to automatically report errors using a pop-up error dialog. (4K)
USE_ZAF_HELP_TIPS	Allow Zinc to display “tool tips” at the mouse cursor when over controls that include a help tip. (5K)
USE_ZAF_KEYBOARD	Allow applications to use a keyboard-like input device. (<1K)
USE_ZAF_MOUSE	Allow applications to use a mouse-like pointing device. (15K)
USE_ZAF_PERSISTENCE	Allow applications to read resources from runtime data files (recommended). Normally needed for multilingual and localized applications, the Zinc help system, and applications developed using Zinc Designer. (up to 85K)
USE_ZAF_RTTI	If your compiler does not include support for RTTI, include this directive to use a Zinc-defined replacement. (6K)
USE_ZAF_SEARCH_PATH	Required to control the locations that Zinc searches for persistence files at runtime. (1K)

PC Compilers

After environment variables have been set (see the previous section, “Configuring your Development Environment”), Zinc 5 libraries may be built. 16-bit Microsoft Windows libraries may be compiled with Borland or Watcom compilers, and 32-bit Microsoft Windows libraries may be compiled with Borland, Watcom, and Microsoft compilers. 16-bit MS-DOS libraries may be compiled with the Borland compiler using the Phar Lap 16-bit extender or Blinker, and 32-bit MS-DOS libraries may be compiled with the Watcom compiler using Watcom’s 32-bit extender. Other compilers such as IBM may work as well but have not been sufficiently tested to be certified by Zinc.

At the command line from the `zaf\source` directory enter one of the following:

```
zmake win16 (builds 16-bit Windows static libraries)
zmake win32 (builds 32-bit Windows static libraries)
```

```
zmake win32dll (builds 32-bit Windows DLL libraries)
zmake dos16 (builds MS-DOS libraries with 16-bit Phar Lap)
zmake dos32 (builds MS-DOS libraries with 32-bit extender)
```

These commands compile and build the libraries and place them in the appropriate location for the compiler specified by the environment variable **ZINC_COMPILER**. For example, if using Borland tools, the completed libraries will be placed in `zaf\lib\borland`. After compilation the following libraries will be created, depending on which of the commands was used:

```
zafw16.lib (16-bit Windows)
zafw16p.lib (16-bit persistence library for Windows)

zafw32.lib (32-bit Windows)
zafw32p.lib (32-bit persistence library for Windows)

zafw32.dll (32-bit Windows DLL, used by "win32dll" executables)
zafw32i.lib (32-bit import library for Windows DLL)
zafw32d.lib (32-bit library which contains WinMain())

zafdl16.lib (MS-DOS with Phar Lap 16-bit extender)
zafdl16p.lib (MS-DOS with Watcom 32-bit extender)
```

Be sure to test your library by running the test application program (.EXE file named according to the build option used) built along with the libraries. This program is found in the current (SOURCE) directory.

In addition to the Zinc 5 library code, Zinc supplies source code to a portable third-party charting library. This library must be built if your application will make use of the `ZafChart` class. See [Third-Party Libraries](#) for more information.

Unix Compilers

After configuring your environment (see the previous section, “Configuring your Development Environment”) you may build Zinc 5 libraries. From the Zinc root directory (usually “ZAF”), simply type “make” at the command line. The XPM library (required for support of the `ZafImage` object on Motif) will be created first, followed by the Zinc libraries. See [Third-Party Libraries](#) for more information on XPM.

After compilation the following libraries will be created and placed in the `zaf/lib` directory:

```
libXpm.a (XPM library)
libZafMtf.a (Archive library for Motif)
libZafMtfP.a (Persistence archive library for Motif)
```

Be sure to test your library by running the “mtest” program built along with the libraries. This program is found in the current (SOURCE) directory.

In addition to the Zinc 5 library code, Zinc supplies source code to a portable third-party charting library. This library must be built if your application will make use of the `ZafChart` class. See [Third-Party Libraries](#) for more information.

pRISM+ and pSOS+

Building libraries and applications for an RTOS can be a complex process. You should be completely familiar with pRISM+ and pSOS+ before proceeding.

1. Select a graphics library.

By default, Zinc is configured to use Zinc's Universal Graphics Library (UGL) when installed for pSOS+.

MetaWINDOW may also be used but Zinc does not supply a version of MetaWINDOW compatible with pSOS+. If using MetaWINDOW, modify the makefile to reference `I_METDSP.CPP` instead of `I_UGLDSP.CPP` and change the compile option `"-DZAF_UGL"` to `"-DZAF_METAWINDOW"`.

2. Select a mouse (pointer) driver.

The mouse driver used when building UGL should be used by the Zinc libraries. To confirm this, examine `zaf/source/p_mouse.cpp`. In the `ZafMouse::OpenMouse()` method, the appropriate UGL pointer device create function should be called and the appropriate header file should be included. If no pointing device is desired, ensure that the macro `USE_ZAF_MOUSE` is not defined (see modularization section.)

3. Build your 2D graphics primitives library, if necessary.

Consult the UGL documentation or other third-party documentation for instructions.

4. Build the Zinc libraries.

Zinc libraries and applications are built from the pRISM+ DOS shell command line. To start this shell, from the menu select "Start | Programs | pRISM+ | Utilities | DOS Prompt x86."

Change to the `zaf/source/psos` directory and type "make". Running make without specifying a target will build a `ram.abs` target, but any of the standard pSOS+ targets can be built by specifying them as the target when running make (e.g., "make `ram.hex`".) The libraries `zafpsos.lib` and `zafpsosp.lib` will be built in the `obj/` directory. A test program will also be built and left in the `zaf/source/psos` directory.

If you encounter "duplicate public symbol" errors while building the Zinc libraries or test program, recheck your development environment configuration against the instructions in the previous chapter. The `"-i-dp"` link option must be defined.

5. Run the library test program.

To run the test program you must download the code to your target system. The best method for downloading to your target depends on your target type and your development environment. In most cases the easiest way to accomplish this is through the pRISM+ debugger on your host. See the pRISM+ documentation for detailed instructions.

Tornado and VxWorks

Building libraries and applications for an RTOS can be a complex process. You should be completely familiar with Tornado and VxWorks before proceeding.

1. Select a graphics library.

By default, Zinc is configured to use Zinc's Universal Graphics Library (UGL) when installed for VxWorks.

MetaWINDOW may also be used but Zinc does not supply a version of MetaWINDOW compatible with VxWorks. If using MetaWINDOW, modify the `gnu.mak` makefile to reference `I_METDSP.CPP` instead of `I_UGLDSP.CPP`, and replace the `"-DZAF_UGL"` compiler option with `"-DZAF_METAWINDOW"`.

2. Select a mouse (pointer) driver.

The mouse driver used when building UGL should be used by the Zinc libraries. To confirm this, examine `zaf/source/v_mouse.cpp`. In the `ZafMouse::OpenMouse()` method the appropriate UGL pointer device create function should be called, and the appropriate header file should be included. If no pointing device is desired, ensure that the macro `USE_ZAF_MOUSE` is not defined (see modularization section.)

3. Build your 2D graphics primitives library, if necessary.

Consult the UGL documentation or other third-party documentation for instructions.

4. Build the Zinc libraries.

Change to the `zaf\source` directory. Type `"make -f gnu.mak"` at the command line. The Zinc libraries `zafVxW.a` and `zafVxWp.a` will be built and copied to the `zaf\lib\gnu` directory. A test application `"vtest.o"` will also be built and left in the `zaf\source` directory.

5. Run the library test program.

You must first download the program and other libraries to your target. Tornado offers various ways of doing this. For example, to download and run the application from the host shell, you would type

```
ld < myapp.o
```

```
myapp
```

After modifying and rebuilding the application, to download and run again you must first unload the existing image. For example,

```
unld "myapp.o"
```

Note: The low-level graphics library may be linked into your application (the default) or downloaded to the target separately prior to downloading the application. Zinc may not be downloaded separately, however, for two reasons.

- The Zinc libraries reference a method (`ZafApplication::Main()`) that is defined by your application.
- Zinc is not currently reentrant. Global variables must be reinitialized before running another Zinc application.

Third-Party Libraries

Zinc provides some third-party libraries that may be compiled and used to provide additional functionality. Other third-party libraries are available that do not ship with Zinc. Contact Zinc for more information if you are interested in using or providing a third-party library for Zinc.

DPC ChartFolio

DPC Technology provides a library that can be used for creating simple charts. This library is not compiled as part of the Zinc libraries and must be compiled separately. This library is required to support the `ZafChart` object.

ChartFolio for PC

To compile the ChartFolio library for PC platforms, invoke `zmake` from the `zaf\cf_zaf\src` directory. Build options are identical to those used for the Zinc 5 libraries: `win16`, `win32`, `win32dll`, `dos16`, and `dos32`.

Following the build process, one of the following libraries (depending on the build option used) will be placed in the same directory as your Zinc 5 libraries.

```
cfzw16.lib (16-bit Windows)
cfzw32.lib (32-bit Windows)
cfzw32.dll and cfzw32i.lib (32-bit DLL Windows)
cfzd16.lib (MS-DOS with Phar Lap 16-bit extender)
cfzd32.lib (MS-DOS with Watcom 32-bit extender)
```

ChartFolio for Unix

To configure the ChartFolio makefiles, first change to the `zaf/cf_zaf` directory and run `minstall`.

To compile the ChartFolio library for Unix platforms, invoke “make” from the `zaf/cf_zaf/src` directory.

Following the build process, `libCfChart.a` will be placed in the same directory as your Zinc 5 libraries.

**XPM Library
(Motif only)**

The source for this library is third-party freeware that Zinc utilizes to provide rich support for the `ZaflImage` class. It is not bundled into the Zinc files, but is shipped as a separate tar file. This tar file, `xpm.tar.Z`, must be downloaded with the rest of the Motif source files. No explicit build instructions are required since the XPM library is built by the same make file as the Zinc libraries (when invoked from the Zinc root directory).

The XPM library is provided on an as-is basis, and its components should not be modified. The “`posix.mak`” file included to build the XPM library is used by the installation script to create “`Makefile`,” but the build options are not modified.

**UGL (Independent
only)**

Zinc has created a highly portable ‘C’ based low-level graphics library for use with Zinc or other high level user interface libraries. UGL provides 2D graphics primitives, hardware interface to video chipsets and input devices, font and OS abstractions, etc.

UGL is primarily used for embedded systems versions of Zinc. `MetaWINDOW` is the default graphics library used by Zinc for DOS.

Building Example Programs

After successfully building the Zinc 5 libraries, it is recommended that you build the Zinc example programs prior to moving on to the “Getting Started” manual where you will actually learn to program using Zinc Application Framework.

Supplied make files allow individual example programs to be built, or related sets of examples (one directory), or all examples at once. Select the build option that is most suitable for you, depending on your strategy for learning Zinc.

PC Compilers

To build all examples at once for the PC, use the `zmakeall` batch file with the appropriate platform option. At the command line from the `zaf\example` directory enter one of the following:

```
zmakeall win16 (builds 16-bit Windows examples)
zmakeall win32 (builds 32-bit Windows examples)
zmakeall win32dll (builds 32-bit Windows examples using the DLL)
zmakeall dos16 (builds MS-DOS examples with 16-bit Phar Lap)
zmakeall dos32 (builds MS-DOS examples with 32-bit extender)
```

To build a related set of example programs (one directory), go to the desired example subdirectory and invoke `zmake`. For example, to build the Hello World examples for 32-bit Microsoft Windows, go to the `zaf/example/hello` directory and type:

```
zmake win32
```

To build a single example program, go to the directory that contains its source code and invoke `zmake` and include the program name on the command line. For example, to build the Hello World 1 example, go to the `zaf/example/hello` directory and type:

```
zmake whllo132.exe
```

To build individual example programs you may need to examine the contents of the make file in each directory for a list of valid program names (targets.)

Unix Compilers

To build all examples at once for Unix, type “make” in the `zaf/example` directory.

To build a related set of example programs (one directory), go to the desired example subdirectory and invoke `make`. For example, to build the Hello World examples, go to the `zaf/example/hello` directory and type:

```
make
```

To build a single example program, go to the directory that contains its source code and invoke make and include the program name on the command line. For example, to build the Hello World 1 example for Motif, go to the `zaf/example/hello` directory and type:

```
make mhello1
```

To build individual example programs you may need to examine the contents of the make file in each directory for a list of valid program names (targets.)

pRISM+ and pSOS+

CadUL makefiles are provided for the `example/hello` programs only. These makefiles serve as an example from which other makefiles may be created (to build other Zinc example programs or custom applications.)

To build the `example/hello` programs for pSOS+, change to the `zaf/example/hello/psos` directory and type “`zmk_psos ram.hex`”. (Any valid standard target may be substituted for “`ram.hex`”.)

`zmk_psos` is a batch file that builds all three hello examples. A batch file is needed in this case because the pSOS+ makefile structure expects the makefile to be called “`makefile`”. Unfortunately, “`makefile`” is capable of building only one application. The batch file simply copies each of three other makefiles (`hello1.mak`, `hello2.mak`, `hello3.mak`) to “`makefile`” and runs make.

`zmk_psos` builds three images called `hello1`, `hello2`, and `hello3`. These may be downloaded to the target and run.

Tornado and VxWorks

A GNU makefile is provided for the `example/hello` programs only. This makefile serves as an example from which other makefiles may be created (to build other Zinc example programs or custom applications.)

To build the `example/hello` programs for VxWorks, change to the `zaf/example/hello` directory and type “`make -fgnu.mak vxworks`”. The three hello examples will be built: `vhello1.o`, `vhello2.o`, and `vhello3.o`. These may be downloaded to the target and run. Refer to the previous chapter (Building Libraries) for specific information about downloading and running on a target.

Learning Zinc Application Framework

Congratulations! You've installed Zinc Application Framework version 5—and along the way you've learned more than you might think. You're now ready to begin studying ZAF application development techniques and working on your own applications.

Zinc provides many tools to help you learn. The most important are excellent documentation and example programs. Both of these ZAF components are being continually expanded to make ZAF as easy to learn and use as possible.

If you have never used electronic documentation, or if you have had a negative experience in the past, you'll be pleasantly surprised by the ZAF versions. Zinc has taken electronic docs to a whole new level. They're easy to use and soon become an invaluable resource. Spend a few minutes familiarizing yourself with the documentation set, and especially the contents of the "Programmer's Reference" manual.

To quickly become productive using Zinc Application Framework we recommend that you step through the following procedure:

- Read the "Getting Started" manual and follow all examples and tutorials. This manual is written for developers at all levels. It begins very slowly by stepping you through sample applications. Then, as the manual progresses it moves more quickly to allow you to think and learn.
- Compile and study the sample programs supplied with ZAF 5. Located in the "zaf/example" directory, these programs demonstrate many of the most fundamental ZAF programming techniques. Building them and understanding their concepts and programming techniques will prepare you to quickly handle common tasks.
- Read a few important sections of the "Programmer's Reference" manual: the "ZafWindowObject" chapter, and the "Event Definitions" and "Property Matrices" appendices.
- Print the "Installation Guide" and "Getting Started" manuals for easier reading, or purchase a set of hard copy documentation from Zinc. Electronic documents are excellent for reference but are more difficult to use while learning.

Enjoy Zinc Application Framework version 5! We look forward to hearing from you as you join the growing crowd of educated developers who rely on ZAF.

Obtaining Technical Support

Free technical support is provided for the installation of Zinc Application Framework. Additional, ongoing technical support is provided to those customers who have purchased Zinc's Customer Assurance Plan (CAP), or on a per-call fee basis.

The following support contact information lists public support access points. CAP members receive separate, private contacts for priority access.

- Public, user community self-support is available via an automated mailing list "zaf5-list@zinc.com". For instructions on subscribing to this and other Zinc lists, send email to "majordomo@zinc.com" and include "help" in the body of the mail, or visit the support section of our web site.
- Direct email support is available via "support@zinc.com". This address is monitored by worldwide support personnel, but is handled only on an as-available resource basis. CAP members receive priority support through another email address.
- Telephone and fax technical support are available world wide from one of Zinc's offices or from a Zinc Partner reseller. Zinc's offices are listed below. Visit the Zinc web site at www.zinc.com for a current list of Zinc Partners.

Zinc Technical Support (North America)

Voice: 1 801-785-8998

FAX: 1 801-785-8996

Email: support@zinc.com

Zinc Technical Support (Europe)

Voice: +44 (0)181 855-9918

FAX: +44 (0)181 316-2211

Email: europe@zinc.com

Zinc Application Framework 5.3

Software License Agreement

Personal Use License - Desktop Applications

There are three different ways to license Zinc Application Framework: (1) Personal Use License for Desktop Applications, (2) Professional Use License for Desktop Applications, and (3) Professional Use License for Embedded Applications. Each license is governed by a different license agreement. The Personal Use License is provided to individuals developing non-commercial, non-distributable, personal-use-only applications. There is no license fee or royalty required for the Personal Use License. However, to exercise rights beyond the Personal Use License, the Developer must purchase a Professional Use License from Zinc, for either Desktop Applications or Embedded Applications, whichever is applicable.

ZINC SOFTWARE INCORPORATED (“ZINC”) IS WILLING TO GRANT YOU THE FOLLOWING LICENSE ON THE CONDITION THAT YOU ACCEPT ALL TERMS OF THIS AGREEMENT (THE “AGREEMENT”).

DO NOT INSTALL, DOWNLOAD OR USE THE ZINC APPLICATION FRAMEWORK SOFTWARE UNTIL YOU HAVE READ AND ACCEPTED THIS LICENSE AGREEMENT. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE YOU ACCEPT THIS LICENSE AGREEMENT. IF YOU DO NOT AGREE TO THIS LICENSE AGREEMENT YOU MUST NOT INSTALL, DOWNLOAD, OR USE THE SOFTWARE.

1. Definitions.

Affiliates. “Affiliates” shall mean Zinc’s parent company, Zinc’s majority-owned subsidiaries, and its and their respective majority-owned subsidiaries.

Desktop Applications. “Desktop Applications” shall mean computer software program applications developed by Licensee which include Linkable Routines and/or Distributable Files and which require a Desktop Operating Environment. Applications which do not require a Desktop Operating Environment are considered to be “Embedded Applications.” A different license must be purchased from Zinc or its authorized reseller for use of the Software to develop Embedded Applications.

Desktop Operating Environment. “Desktop Operating Environment” shall mean one or more of the following: Microsoft Windows; Microsoft Windows NT; IBM OS/2; Apple MacOS; and UNIX.

Developer. “Developer” shall mean the individual natural person who accepts and agrees to this Agreement. No corporation, partnership, limited liability company or other organization, government or business entity may be a Developer under this Agreement. They may, however, purchase Professional Use Licenses from Zinc or its authorized reseller. Developer is the only person authorized to use the Software under the terms of this Agreement.

Distributable Files. “Distributable Files” shall mean those files identified as distributable in the Documentation.

Documentation. “Documentation” shall mean the online documentation and printed documentation, if any, provided to Developer in connection with this Agreement.

Licensed Source Code. “Licensed Source Code” shall mean that portion of the Software’s source code which is provided to Developer in connection with this Agreement.

Linkable Routines. “Linkable Routines” shall mean the object code routines in the Software libraries (e.g., *.LIB, lib*.a).

Non-commercial Desktop Applications. “Non-commercial Desktop Applications” shall mean Desktop Applications developed by Developer that: (a) are for use by Developer only, and not for use by, or distribution to, any employer, customer or other third party. A different license must be purchased from Zinc or its authorized reseller for use of the Software to develop Desktop Applications for commercial purposes or use.

Software. “Software” shall mean the Zinc Application Framework computer programs and Documentation provided with this Agreement. Software includes without limitation Linkable Routines, Distributable Files, non-distributable files, and Licensed Source Code. Any modifications or additions to the Software created by Developer pursuant to the license granted hereunder shall belong solely to Zinc, and shall be considered part of Software pursuant to the license granted herein.

2. License. Subject to the other provisions of this Agreement, Zinc grants to Developer a restricted, nonexclusive, nontransferable license (without the right to grant sublicenses) for one person (the Developer):

(a) to use the Software to develop Non-commercial Desktop Applications (as defined above); and

(b) to incorporate Linkable Routines and/or Distributable Files into Non-commercial Desktop Applications developed by Developer, provided that the Linkable Routines and/or Distributable Files have been incorporated in such a way that they cannot be used apart from the Non-commercial Desktop Applications; and

(c) to use such Linkable Routines and/or Distributable Files as part of or with the Non-commercial Desktop Applications, pursuant to the license granted hereunder but not separate from such Non-commercial Desktop Applications; and

(d) to use the Licensed Source Code to maintain and modify the Software to conform with Developer’s needs in creating Non-commercial Desktop Applications.

The foregoing license does not include any right to use the Software to develop any computer programs that are competitive with, or that can be used in lieu of, the Software. Rights not expressly granted to Developer in this Agreement are reserved by Zinc. Developer acknowledges that Zinc (and its licensor(s), if applicable) own the copyrights and other intellectual property in and to the Software.

3. Copies. Developer may make copies of the Software provided that each such copy: (a) is created as an essential step in the utilization of the Software in accordance with this Agreement, or (b) is only for archival purposes to back-up the licensed use of the Software. Developer may also make copies of the Software to the extent reasonably needed to exercise rights under this Agreement, including without limitation those rights described in Section 4 (“Distribution Rights”). All Zinc proprietary notices must be faithfully reproduced and included on copies made by Developer, or as otherwise directed from time to time by Zinc. Developer may not make any other copies of the Software. If printable, the online Documentation may be printed by Developer and used by Developer, but only in connection with the licensed use of the Software.

4. Distribution Rights. A copy of the Software in its complete and unmodified form as provided by Zinc may be distributed or transferred by Developer to any other individual natural person solely and exclusively pursuant to the terms of this Agreement as accompany such copy. Except as provided herein, Linkable Routines and Distributable Files shall not be distributed or transferred by Developer other than as part of the Software and shall in no event be distributed or transferred as part of or with any Non-commercial Desktop Application. Developer may distribute, transfer, and disclose Developer’s Source Code in Non-commercial Desktop Applications, provided that no part of the Licensed Source Code (including any modifications or derivative works thereof) is distributed, transferred, or disclosed.

5. Protection Of The Software. Except as expressly authorized in this Agreement, Developer may not: (a) disassemble, decompile or otherwise reverse engineer the Software, or (b) create derivative works based upon the Software, or (c) rent, lease, sublicense, distribute, transfer, copy, reproduce, or timeshare the Software, or (d) modify the Software (including any deletion of code from or addition of code to the Software) or (e) allow any third party to access or use the Software, or to do any of the foregoing. The Software is the valuable proprietary information of Zinc (or its licensor(s), if applicable), and may not be used or disclosed except as expressly permitted by this Agreement. The Licensed Source Code (whether modified or unmodified) may not be disclosed or distributed by Developer to any other person.

6. Disclaimer Regarding Modified Software. Zinc and its Affiliates accept no responsibility or obligation to maintain or support any unmodified or modified versions of the Software and no warranties are applicable to any modified versions. There is no warranty that the Software is suitable for modification and all modifications are undertaken at the risk and discretion of Developer.

7. Limited Warranty. Zinc warrants that the media on which Software is delivered will be free from defects in materials or workmanship for a period of ninety (90) days from the date on which Developer receives such media. If during the foregoing ninety (90) day warranty period the media on which Software is delivered proves to be defective and if it is returned to Zinc (postage prepaid), Zinc will

repair or replace such media, at Zinc’s option, and as Developer’s sole remedy for any breach of warranty hereunder.

8. Disclaimer And Limited Liability.

(a) Disclaimer Of Warranties. OTHER THAN THE LIMITED WARRANTY STATED ABOVE, ZINC AND ITS AFFILIATES MAKE NO WARRANTY OF ANY KIND AND THE SOFTWARE IS LICENSED AND PROVIDED TO LICENSEE STRICTLY ON AN “AS IS” BASIS. ZINC AND ITS AFFILIATES DISCLAIM AND EXCLUDE ALL WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ZINC AND ITS AFFILIATES DO NOT WARRANT THAT THE SOFTWARE WILL SATISFY LICENSEE’S REQUIREMENTS OR THAT IT IS WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED.

(b) Limitation On Liability. ZINC AND ITS AFFILIATES WILL NOT BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF USE, INTERRUPTION OF BUSINESS, NOR FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, WHETHER UNDER THIS AGREEMENT OR OTHERWISE, IRRESPECTIVE OF WHETHER ZINC HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL THE LIABILITY OF ZINC OR ITS AFFILIATES HEREUNDER EXCEED FIFTY U.S. DOLLARS.

9. Termination. The Agreement and all licenses granted by Zinc hereunder shall automatically terminate if Developer materially breaches this Agreement. Upon termination of the Agreement, Developer shall cease all use of the Software and shall destroy all copies of the Software within the possession or control of Developer and shall return the original Software media and Documentation to Zinc. In the event of any termination of this Agreement, the provisions of Section 1 (“Definitions”), Section 8 (“Disclaimer and Limited Liability”), Section 9 (“Termination”), Section 10 (“Export Laws”), Section 11 (“Severability”), Section 12 (“Governing Law”) and Section 13 (“Entire Agreement”) shall survive and continue to bind the parties.

10. Export Laws. Developer shall not export, disclose or distribute any Software in violation of any applicable laws or regulations, including the export laws and regulations of the United States.

11. Severability. If any provision in this Agreement is invalid or unenforceable or contrary to applicable law, such provision shall be construed, limited, or altered, as necessary, to eliminate the invalidity or unenforceability or the conflict with applicable law, and all other provisions of this Agreement shall remain in effect.

12. Governing Law. This Agreement shall be governed by the laws of the State of California and the United States of America without giving effect to conflict or choice of law principles. The United Nations Convention on Contracts for the International Sale of Goods shall not be applicable and is rejected by the parties. The State and Federal Courts located in San Francisco County, California, shall have sole jurisdiction over any disputes arising under this Agreement and the parties hereby submit to the jurisdiction of such courts.

13. Entire Agreement. This Agreement is the entire agreement between the parties regarding this subject matter and supersedes all prior discussions, negotiations, agreements, and the like. This Agreement may be modified only in writing signed by authorized representatives of both parties. No vendor, distributor, dealer, reseller, sales person or other person is authorized by Zinc to modify this Agreement or to

make any warranty, representation or promise which is different than, or in addition to, the warranties, representations or promises of this Agreement. Any purported oral modification of this Agreement shall be void.

Zinc Application Framework 5.3

Software License Agreement

Professional Use License - Desktop Applications

There are three different ways to license Zinc Application Framework: (1) Personal Use License for Desktop Applications, (2) Professional Use License for Desktop Applications, and (3) Professional Use License for Embedded Applications. Each license is governed by a different license agreement. The Personal Use License is provided to individuals developing non-commercial, non-distributable, personal-use-only applications. There is no license fee or royalty required for the Personal Use License. However, to exercise rights beyond the Personal Use License, the Licensee must purchase a Professional Use License from Zinc, for either Desktop Applications or Embedded Applications, whichever is applicable.

THIS IS A PROFESSIONAL USE LICENSE AGREEMENT FOR DESKTOP APPLICATIONS (“AGREEMENT”) BETWEEN ZINC SOFTWARE INCORPORATED (“ZINC”) AND THE ENTITY OR INDIVIDUAL (“LICENSEE” OR “YOU”) WHO HAS RECEIVED THIS AGREEMENT AND THE ACCOMPANYING ZINC APPLICATION FRAMEWORK SOFTWARE AND DOCUMENTATION, AND PAID THE ASSOCIATED FEES TO ZINC. ZINC IS WILLING TO GRANT TO LICENSEE THE FOLLOWING LICENSE ON THE CONDITION THAT LICENSEE ACCEPTS ALL TERMS OF THIS AGREEMENT.

DO NOT INSTALL, DOWNLOAD OR USE THE ZINC APPLICATION FRAMEWORK SOFTWARE UNTIL YOU HAVE READ AND ACCEPTED THIS LICENSE AGREEMENT. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE YOU ACCEPT THIS LICENSE AGREEMENT. IF YOU DO NOT AGREE TO THIS LICENSE AGREEMENT: (A) YOU MUST NOT INSTALL, DOWNLOAD, OR USE THE SOFTWARE, AND (B) YOU MAY RETURN THE SOFTWARE, INCLUDING ALL PACKAGING, MEDIA, AND DOCUMENTATION, FOR A REFUND, PROVIDED THAT THE RETURN IS MADE WITHIN TEN DAYS OF THE DATE OF PURCHASING THIS LICENSE.

1. Definitions.

Affiliates. “Affiliates” shall mean Zinc’s parent company, Zinc’s majority-owned subsidiaries, and its and their respective majority-owned subsidiaries.

Desktop Applications. “Desktop Applications” shall mean computer software program applications developed by Licensee which include Linkable Routines and/or Distributable Files and which require a Desktop Operating Environment. Applications which do not require a Desktop Operating Environment are considered to be “Embedded Applications.” A different license must be purchased from Zinc or its authorized reseller for use of the Software to develop Embedded Applications.

Desktop Operating Environment. “Desktop Operating Environment” shall mean one or more of the following: Microsoft Windows; Microsoft Windows NT; IBM OS/2; Apple MacOS; and UNIX.

Developer. “Developer” shall mean the individual natural person who accepts and agrees to this Agreement. If Developer is an employee of a company and intends to use the Software within the scope of his/her employment, then acceptance of this Agreement is also made on behalf of the company. The Developer is the only person authorized to use the Software under the terms of this Agreement.

Distributable Files. “Distributable Files” shall mean those files identified as distributable in the Documentation.

Documentation. “Documentation” shall mean the online documentation and printed documentation, if any, provided to Licensee in connection with this Agreement.

Error. “Error” shall mean any failure of the Software as delivered by Zinc to conform to its published specifications during ordinary use in accordance with this Agreement.

Licensed Source Code. “Licensed Source Code” shall mean that portion of the Software’s source code which is provided to Licensee in connection with this Agreement.

Linkable Routines. “Linkable Routines” shall mean the object code routines in the Software libraries (e.g. *.LIB, lib*.a).

Optional Modules. “Optional Modules” shall mean an optional component of the Software as designated by Zinc (e.g. platform modules, Unicode support module).

Required Module. “Required Module” shall mean the required component of the Software as designated by Zinc (i.e. “Shared Code”).

Software. “Software” shall mean the Zinc Application Framework computer programs and Documentation provided with this Agreement. Software consists of the Required Module and one or more Optional Modules and these include, without limitation, Linkable Routines, Distributable Files, non-distributable files, and Licensed Source Code. Any modifications or additions to the Software created by Licensee pursuant to the license granted hereunder shall belong solely to Zinc, and shall be considered part of Software pursuant to the license granted herein.

2. Development Rights.

(a) License. Subject to the other provisions of this Agreement, including without limitation the payment by Licensee all license fees, Zinc grants to Licensee a restricted, nonex-

clusive, nontransferable license (without the right to grant sublicenses) for one person (the Developer):

(i) to use the Software to develop Desktop Applications (as defined above);

(ii) to incorporate Linkable Routines and/or Distributable Files into Desktop Applications developed by Developer, provided that the Linkable Routines and/or Distributable Files have been incorporated in such a way that they cannot be used apart from the Desktop Applications; and

(iii) to use Linkable Routines and/or Distributable Files as part of or with Desktop Applications, pursuant to the license granted hereunder but not separate from such Desktop Applications;

(iv) to use the Licensed Source Code to maintain and modify the Software to conform with Licensee's needs in creating Desktop Applications.

(b) Restrictions.

(i) Required Module. The Developer must obtain a license for his or her own Required Module. The Required Module may be used by the Developer only on any computer on which at least one of the same Developer's Optional Modules is used as permitted herein.

(ii) Optional Modules. The Developer must obtain a license for at least one Optional Module designed for a particular operating system platform and computer type. Each Optional Module may be used by the Developer only on a single computer running under the operating system designated on a license certificate or invoice for the Optional Module. Licensee may not use an Optional Module on more than one computer at any given time unless an additional license for each additional computer is purchased. Notwithstanding anything in this Agreement to the contrary, this license does not include, and Licensee has no right to install, use or copy, any Optional Module not designated on a license certificate or invoice. If Licensee desires to use additional Optional Modules, a license for such additional Optional Modules must first be purchased from Zinc or its authorized reseller. Additional Optional Modules for which a license is purchased shall be governed by this Agreement and shall be deemed part of the Software.

(iii) Other Restrictions. The Software is licensed on a per Developer basis. The Developer may not be changed, without the written consent of Zinc. Licenses for additional Developers must be purchased from Zinc or its authorized resellers. The foregoing license does not include any right to use the Software to develop any computer programs that are competitive with, or that can be used in lieu of, the Software. Rights not expressly granted to Licensee in this Agreement are reserved by Zinc. Licensee acknowledges that Zinc (and its licensor(s), if applicable) own the copyrights and other intellectual property in and to the Software.

3. Copies. Licensee may make copies of the Software provided that each such copy: (a) is created as an essential step in the utilization of the Software in accordance with this Agreement, or (b) is only for archival purposes to back-up the licensed use of the Software. Licensee may also make copies of the Software to the extent reasonably needed to exercise those rights described in Section 4 ("Distribution License"). All Zinc proprietary notices must be faithfully reproduced and included on copies made by Licensee. Licensee may not make any other copies of the Software. The

online Documentation may be printed by Licensee and used by Licensee, but only in connection with the licensed use of the Software.

4. Distribution License. Subject to the other provisions of this Agreement, Zinc grants to Licensee a restricted, nonexclusive, nontransferable license (without the right to grant sublicenses) which includes the following distribution rights:

(a) authorization for Licensee to incorporate Linkable Routines into and/or distribute Distributable Files with Desktop Applications developed by Licensee and to distribute them as part of such Desktop Applications to Licensee's customers, provided that the Linkable Routines and/or Distributable Files have been incorporated in such a way that they cannot be used apart from the Desktop Applications,

(b) authorization for Licensee to license Licensee's customers to use such Linkable Routines and/or Distributable Files as part of the Desktop Applications, but not separate from such Desktop Applications.

5. Distribution Restrictions. Except for the Linkable Routines and Distributable Files, no portion of the Software may be distributed or transferred by Licensee. The Linkable Routines and Distributable Files may not be distributed as part of any computer program other than a Desktop Application as defined above without the express written permission of Zinc. Licensee must include an appropriate Zinc copyright notice, in accordance with guidelines published by Zinc from time to time, on all copies of Licensee's Desktop Applications in which Linkable Routines are incorporated or with which Distributable Files are distributed. Third parties who receive any Linkable Routines or Distributable Files as part of a Desktop Application under Section 4 ("Distribution License") may not use such Linkable Routines or Distributable Files for any development purposes. Licensee shall ensure that its agreements with such third parties are consistent with this Agreement and are no less protective of Zinc, its Affiliates and licensors as the restrictions listed in Sections 6 ("Protection of the Software"), 10 ("Disclaimer and Limited Liability"), 12b ("Distribution of Desktop Applications to U.S. Government End Users"), and 13 ("Export Laws").

6. Protection Of The Software. Except as expressly authorized in this Agreement, Licensee may not: (a) disassemble, decompile or otherwise reverse engineer the Software, or (b) create derivative works based upon the Software, or (c) rent, lease, sublicense, distribute, transfer, copy, reproduce, or timeshare the Software, or (d) modify the Software (including any deletion of code from or addition of code to the Software) or (e) allow any third party to access or use the Software, or to do any of the foregoing. The Software is the valuable proprietary information of Zinc (or its licensor(s), if applicable), and may not be used or disclosed except as expressly permitted by this Agreement. The Licensed Source Code (whether modified or unmodified) may not be disclosed or distributed by Licensee to any other person.

7. Disclaimer Regarding Modified Software. Zinc and its Affiliates accept no responsibility or obligation to maintain or support modified versions of the Software and no warranties are applicable to such modified versions. There is no warranty that the Software is suitable for modification and all modifications are undertaken at the risk and discretion of Licensee.

8. Maintenance.

(a) Provision of Maintenance. In the event that Licensee has agreed to receive Maintenance as indicated on a license certificate or invoice, and has paid all associated fees as invoiced by Zinc, Licensee shall be entitled to receive Maintenance as hereinafter described. For purposes of this Agreement, "Maintenance" shall consist of:

(i) The right of the Developer to contact Zinc by telephone or e-mail and to consult with Zinc regarding the installation, functions and operation of the unmodified Software;

(ii) The right of the Developer to contact Zinc by telephone or e-mail to consult with Zinc regarding Errors in the unmodified Software which Errors have been brought to Zinc's attention by the Developer;

(iii) The right to obtain from Zinc certain updates for the Software from time to time as such updates are made generally available by Zinc.

(b) Hours and Facilities. Zinc shall make Maintenance available only during Zinc's normal business days and normal business hours.

(c) Limitations. Licensee must elect to receive Maintenance on either all or none of the Software, and may not elect to receive Maintenance on some Software and not on other Software. Zinc shall have no obligation to provide Maintenance for any modified Software or any Software which is not the most recent version or release or for which Zinc has provided updates which have not been applied by Licensee to the Software for a period of more than one (1) year from the date said updates were made generally available. Any failure of Licensee to pay all Maintenance fees as invoiced by Zinc shall, at Zinc's sole option, immediately and permanently eliminate any obligation of Zinc to provide Maintenance hereunder.

9. Limited Warranty.

(a) Media And Documentation. Zinc warrants that the media on which Software is delivered will be free from defects in materials or workmanship for a period of ninety (90) days from the date on which Licensee receives such media. If during the foregoing ninety (90) day warranty period the media on which Software is delivered proves to be defective and if it is returned to Zinc (postage prepaid), Zinc will repair or replace such media, at Zinc's option, and as Licensee's sole remedy for any breach of warranty hereunder.

(b) Software. Zinc warrants the Software substantially conforms to the specifications in the Documentation and if the nonconformity is reported in writing by Licensee to Zinc within 90 days from the date the license is purchased, then Zinc shall either remedy the nonconformity or offer to refund the purchase price to Licensee upon a return of all copies of the Software (including all packaging, media, and Documentation) to Zinc. In the event of a refund, the Agreement shall terminate.

10. Disclaimer And Limited Liability.

(a) Disclaimer Of Warranties. OTHER THAN THE LIMITED WARRANTIES STATED ABOVE, ZINC AND ITS AFFILIATES MAKE NO WARRANTY OF ANY KIND AND THE SOFTWARE IS LICENSED AND PROVIDED TO LICENSEE STRICTLY ON AN "AS IS" BASIS. ZINC AND ITS AFFILIATES DISCLAIM AND EXCLUDE ALL WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRAN-

TIES OF NONINFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ZINC AND ITS AFFILIATES DO NOT WARRANT THAT THE SOFTWARE WILL SATISFY LICENSEE'S REQUIREMENTS OR THAT IT IS WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED.

(b) Limitation On Liability. ZINC AND ITS AFFILIATES WILL NOT BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF USE, INTERRUPTION OF BUSINESS, NOR FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, WHETHER UNDER THIS AGREEMENT OR OTHERWISE, IRRESPECTIVE OF WHETHER ZINC HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL THE LIABILITY OF ZINC OR ITS AFFILIATES HEREUNDER EXCEED THE TOTAL PAYMENT MADE BY LICENSEE TO PURCHASE THIS LICENSE.

11. Termination. The Agreement and all licenses granted by Zinc hereunder shall automatically terminate if Licensee materially breaches this Agreement. Upon termination of the Agreement, Licensee shall cease all use of the Software and shall destroy all copies of the Software within the possession or control of Licensee and shall return the original Software media and Documentation to Zinc. In the event of any termination of this Agreement, the provisions of Section 1 ("Definitions"), Section 9 ("Disclaimer and Limited Liability"), Section 11 ("Termination"), Section 13 ("Export Laws"), Section 14 ("Severability"), Section 15 ("Governing Law") and Section 16 ("Entire Agreement") shall survive and continue to bind the parties.

12. Government End Users. A "U.S. Government End User" shall mean any agency or entity of the government of the United States. The following shall apply:

(a) If Licensee is U.S. Government End User. If Licensee is a U.S. Government End User, then this Subsection (a) shall apply. The Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire the Software with only those rights set forth herein. The Software (including related documentation) is provided to U.S. Government End Users: (i) only as a commercial end item; and (ii) only pursuant to this Agreement.

(b) Distribution of Desktop Applications to U.S. Government End Users. When distributing or licensing a Desktop Application to a U.S. Government End User, Licensee shall identify the Software or modified Software in the Desktop Application as a "commercial item," as that term is defined at 48 C.F.R. 2.101 (Oct. 1995), and more specifically shall be identified as "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), Licensee will provide the Software or modified Software in any Desktop Application (including related documentation) to U.S. Government End Users: (i) only as a commercial end item; and (ii) only pursuant to the

end user license agreement used by Licensee to license the Desktop Application to the U.S. Government End User.

13. Export Laws. Licensee shall not export, disclose or distribute any Software in violation of any applicable laws or regulations, including the export laws and regulations of the United States.

14. Severability. If any provision in this Agreement is invalid or unenforceable or contrary to applicable law, such provision shall be construed, limited, or altered, as necessary, to eliminate the invalidity or unenforceability or the conflict with applicable law, and all other provisions of this Agreement shall remain in effect.

15. Governing Law. This Agreement shall be governed by the laws of the State of California and the United States of America without giving effect to conflict or choice of law principles. The United Nations Convention on Contracts for

the International Sale of Goods shall not be applicable and is rejected by the parties. The State and Federal Courts located in San Francisco County, California, shall have sole jurisdiction over any disputes arising under this Agreement and the parties hereby submit to the jurisdiction of such courts.

16. Entire Agreement. This Agreement is the entire agreement between the parties regarding this subject matter and supersedes all prior discussions, negotiations, agreements, and the like. This Agreement may be modified only in writing signed by authorized representatives of both parties. No vendor, distributor, dealer, reseller, sales person or other person is authorized by Zinc to modify this Agreement or to make any warranty, representation or promise which is different than, or in addition to, the warranties, representations or promises of this Agreement. Any purported oral modification of this Agreement shall be void.

Zinc Application Framework 5.3

Software License Agreement

Professional Use License - Embedded Applications

There are three different ways to license Zinc Application Framework: (1) Personal Use License for Desktop Applications, (2) Professional Use License for Desktop Applications, and (3) Professional Use License for Embedded Applications. Each license is governed by a different license agreement. The Personal Use License is provided to individuals developing non-commercial, non-distributable, personal-use-only applications. There is no license fee or royalty required for the Personal Use License. However, to exercise rights beyond the Personal Use License, the Licensee must purchase a Professional Use License from Zinc, for either Desktop Applications or Embedded Applications, whichever is applicable.

THIS IS A PROFESSIONAL USE LICENSE AGREEMENT FOR EMBEDDED APPLICATIONS (“AGREEMENT”) BETWEEN ZINC SOFTWARE INCORPORATED (“ZINC”) AND THE ENTITY OR INDIVIDUAL (“LICENSEE” OR “YOU”) WHO HAS RECEIVED THIS AGREEMENT AND THE ACCOMPANYING ZINC APPLICATION FRAMEWORK SOFTWARE AND DOCUMENTATION, AND PAID THE ASSOCIATED FEES TO ZINC. ZINC IS WILLING TO GRANT TO LICENSEE THE FOLLOWING LICENSE ON THE CONDITION THAT LICENSEE ACCEPTS ALL TERMS OF THIS AGREEMENT.

DO NOT INSTALL, DOWNLOAD OR USE THE ZINC APPLICATION FRAMEWORK SOFTWARE UNTIL YOU HAVE READ AND ACCEPTED THIS LICENSE AGREEMENT. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE YOU ACCEPT THIS LICENSE AGREEMENT. IF YOU DO NOT AGREE TO THIS LICENSE AGREEMENT: (A) YOU MUST NOT INSTALL, DOWNLOAD, OR USE THE SOFTWARE, AND (B) YOU MAY RETURN THE SOFTWARE, INCLUDING ALL PACKAGING, MEDIA, AND DOCUMENTATION, FOR A REFUND, PROVIDED THAT THE RETURN IS MADE WITHIN TEN DAYS OF THE DATE OF PURCHASING THIS LICENSE.

1. Definitions.

Affiliates. “Affiliates” shall mean Zinc’s parent company, Zinc’s majority-owned subsidiaries, and its and their respective majority-owned subsidiaries.

Desktop Operating Environment. “Desktop Operating Environment” shall mean one or more of the following: Microsoft Windows; Microsoft Windows NT; IBM OS/2; Apple MacOS; and UNIX.

Developer. “Developer” shall mean the individual natural person who accepts and agrees to this Agreement. If Developer is an employee of a company and intends to use the Software within the scope of his/her employment, then acceptance of this Agreement is also made on behalf of the company. The Developer is the only person authorized to use the Host Software under the terms of this Agreement.

Distributable Files. “Distributable Files” shall mean those files identified as distributable in the Documentation.

Documentation. “Documentation” shall mean the online documentation and printed documentation, if any, provided to Licensee in connection with this Agreement.

Embedded Application. “Embedded Application” shall mean computer software program applications developed by Licensee which include Linkable Routines and/or Distributable Files and which do not require a Desktop Operating Environment. Applications which require a Desktop Operating Environment are considered to be “Desktop Applications.” A different license must be purchased from Zinc or its authorized reseller for use of the Software to develop Desktop Applications.

Error. “Error” shall mean any failure of the Software as delivered by Zinc to conform to its published specifications during ordinary use in accordance with this Agreement.

Host Platform. “Host Platform” shall mean a computer platform on which the Software is used in the development of the Licensed Embedded Application.

Host Software. “Host Software” shall mean (a) all components of the Software which are able to execute on any platform other than the Target Platform; or (b) Software which is able to execute on the Target Platform other than Linkable Routines or Distributable Files.

Licensed Embedded Application. “Licensed Embedded Application” shall mean the Embedded Application specified on the Project Description Form and is limited to the Target Platform. This License applies only to one Licensed Embedded Application. Additional licenses must be purchased from Zinc or its authorized reseller for use of the Software in the development of other Embedded Applications.

Licensed Source Code. “Licensed Source Code” shall mean that portion of the Software’s source code which is provided to Licensee in connection with this Agreement.

Linkable Routines. “Linkable Routines” shall mean the object code routines in the Software libraries (e.g., *.LIB, lib*.a).

Project. “Project” shall mean a Licensee development effort, the purpose and result of which are the creation of a Licensed Embedded Application. This Agreement contains a license for a single Project consisting of the use of the Software on a Host Platform for the development of a

Licensed Embedded Application which will be used on a Target Platform.

Project Description Form. “Project Description Form” shall mean a written or electronic form signed by Zinc and Licensee, which describes the Target Platform, the Licensed Embedded Application and the name of the Project.

Runtime License. “Runtime License” shall mean a royalty-bearing license (to be granted by Zinc to Licensee under a separate license agreement) to distribute one copy of the Licensed Embedded Application deployed on a Target Platform. The number of Runtime Licenses purchased by Developer will be indicated on the license certificate or invoice issued by Zinc.

Software. “Software” shall mean the Zinc Application Framework computer programs, the Universal Graphics Library (“UGL”) computer programs, and Documentation provided with this Agreement. Software consists of Target Software and Host Software and these include, without limitation, Linkable Routines, Distributable Files, non-distributable files, and Licensed Source Code. Any modifications or additions to the Software created by Licensee pursuant to the license granted hereunder shall belong solely to Zinc, and shall be considered part of Software pursuant to the license granted herein.

Target Platform. “Target Platform” shall mean the hardware and software platform on which the Licensed Embedded Application will run. The license is limited to the Target Platform specified in the Project Description Form.

Target Software. “Target Software” shall mean the Linkable Routines and the Distributable Files which are utilized on the Target Platform, and which may be included in the Licensed Embedded Application.

2. Project Development License. Subject to the other provisions of this Agreement, including without limitation the payment by Licensee of all license fees, Zinc grants to Licensee a restricted, nonexclusive, nontransferable license (without the right to grant sublicenses):

(a) to use the Target Software on the Target Platform and a Host Platform (without limitation on the number of computers or workstations) to develop the Licensed Embedded Application; and

(b) to use the Host Software on a Host Platform to develop the Licensed Embedded Application; and

(c) to use the Licensed Source Code to modify the Software to conform with Developer’s needs in creating the Licensed Embedded Application.

3. Restrictions. Host Software is licensed on a per Developer, per Project, per Embedded Application basis. Host Software may not be shared or transferred from one Developer to another. Target Software is licensed on a per Project and per Embedded Application basis. The Project or Licensed Embedded Application description contained on the Project Description Form may not be changed without the written consent of Zinc. Licenses for additional Projects, Developers, or Target Platforms must be purchased from Zinc or its authorized resellers. The foregoing license does not include any right to use the Software to develop any computer programs that are competitive with, or that can be used in lieu of, the Software. Rights not expressly granted to Licensee in this Agreement are reserved by Zinc. Licensee acknowledges that Zinc (and its licensor(s), if applicable)

own the copyrights and other intellectual property in and to the Software. THIS LICENSE DOES NOT INCLUDE RUNTIME LICENSES OR PROVIDE THE RIGHT TO DISTRIBUTE ANY PORTION OF THE SOFTWARE IN LICENSED EMBEDDED APPLICATIONS TO THIRD PARTIES. Pricing and licenses for Runtime License distribution are available from Zinc and its authorized resellers.

4. Copies. Licensee may make copies of the Software provided that each such copy: (a) is created as an essential step in the utilization of the Software in accordance with this Agreement, or (b) is only for archival purposes to back-up the licensed use of the Software. All Zinc proprietary notices must be faithfully reproduced and included on copies made by Licensee. Licensee may not make any other copies of the Software. The online Documentation may be printed by Licensee and used by Licensee, but only in connection with the licensed use of the Software.

5. Protection Of The Software. Except as expressly authorized in this Agreement, Licensee may not: (a) disassemble, decompile or otherwise reverse engineer the Software, or (b) create derivative works based upon the Software, or (c) rent, lease, sublicense, distribute, transfer, copy, reproduce, or timeshare the Software, or (d) modify the Software (including any deletion of code from or addition of code to the Software) or (e) allow any third party to access or use the Software, or to do any of the foregoing. The Software is the valuable proprietary information of Zinc (or its licensor(s), if applicable), and may not be used or disclosed except as expressly permitted by this Agreement. The Licensed Source Code (whether modified or unmodified) may not be disclosed or distributed by Licensee to any other person.

6. Disclaimer Regarding Modified Software. Zinc and its Affiliates accept no responsibility or obligation to maintain or support modified versions of the Software and no warranties are applicable to such modified versions. There is no warranty that the Software is suitable for modification and all modifications are undertaken at the risk and discretion of Licensee.

7. Maintenance.

(a) Provision of Maintenance. In the event that Licensee has agreed to receive Maintenance as indicated on a license certificate or invoice, and has paid all associated fees as invoiced by Zinc, Licensee shall be entitled to receive Maintenance as hereinafter described. For purposes of this Agreement, “Maintenance” shall consist of:

(i) The right of the Developer to contact Zinc by telephone or e-mail and to consult with Zinc regarding the installation, functions and operation of the unmodified Software;

(ii) The right of the Developer to contact Zinc by telephone or e-mail to consult with Zinc regarding Errors in the unmodified Software which Errors have been brought to Zinc’s attention by the Developer;

(iii) The right to obtain from Zinc certain updates for the Software from time to time as such updates are made generally available by Zinc.

(b) Hours and Facilities. Zinc shall make Maintenance available only during Zinc’s normal business days and normal business hours.

(c) Limitations. Licensee must elect to receive Maintenance on either all or none of the Software, and may not elect to receive Maintenance on some Software and not on other

Software. Zinc shall have no obligation to provide Maintenance for any modified Software or any Software which is not the most recent version or release or for which Zinc has provided updates which have not been applied by Licensee to the Software for a period of more than one (1) year from the date said updates were made generally available. Any failure of Licensee to pay all Maintenance fees as invoiced by Zinc shall, at Zinc's sole option, immediately and permanently eliminate any obligation of Zinc to provide Maintenance hereunder.

8. Limited Warranty.

(a) Media And Documentation. Zinc warrants that the media on which Software is delivered will be free from defects in materials or workmanship for a period of ninety (90) days from the date on which Licensee receives such media. If during the foregoing ninety (90) day warranty period the media on which Software is delivered proves to be defective and if it is returned to Zinc (postage prepaid), Zinc will repair or replace such media, at Zinc's option, and as Licensee's sole remedy for any breach of warranty hereunder.

(b) Software. Zinc warrants the Software substantially conforms to the specifications in the Documentation and if the nonconformity is reported in writing by Licensee to Zinc within 90 days from the date the license is purchased, then Zinc shall either remedy the nonconformity or offer to refund the purchase price to Licensee upon a return of all copies of the Software (including all packaging, media, and Documentation) to Zinc. In the event of a refund, the Agreement shall terminate.

9. Disclaimer And Limited Liability.

(a) **Disclaimer Of Warranties.** OTHER THAN THE LIMITED WARRANTIES STATED ABOVE, ZINC AND ITS AFFILIATES MAKE NO WARRANTY OF ANY KIND AND THE SOFTWARE IS LICENSED AND PROVIDED TO LICENSEE STRICTLY ON AN "AS IS" BASIS. ZINC AND ITS AFFILIATES DISCLAIM AND EXCLUDE ALL WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. ZINC AND ITS AFFILIATES DO NOT WARRANT THAT THE SOFTWARE WILL SATISFY LICENSEE'S REQUIREMENTS OR THAT IT IS WITHOUT DEFECT OR ERROR OR THAT THE OPERATION THEREOF WILL BE UNINTERRUPTED.

(b) **Limitation On Liability.** ZINC AND ITS AFFILIATES WILL NOT BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF USE, INTERRUPTION OF BUSINESS, NOR FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, WHETHER UNDER THIS AGREEMENT OR OTHERWISE, IRRESPECTIVE OF WHETHER ZINC HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL THE LIABILITY OF ZINC OR ITS AFFILIATES HEREUNDER EXCEED THE TOTAL PAYMENT MADE BY LICENSEE TO PURCHASE THIS LICENSE.

10. Termination. The Agreement and all licenses granted by Zinc hereunder shall automatically terminate if Licensee materially breaches this Agreement. Upon termination of the Agreement, Licensee shall cease all use of the Software and shall destroy all copies of the Software within the possession or control of Licensee and shall return the original Software media and Documentation to Zinc. In the event of any termination of this Agreement, the provisions of Section 1 ("Definitions"), Section 9 ("Disclaimer and Limited Liability"), Section 10 ("Termination"), Section 12 ("Export Laws"), Section 13 ("Severability"), Section 14 ("Governing Law") and Section 15 ("Entire Agreement") shall survive and continue to bind the parties.

11. Government End Users. A "U.S. Government End User" shall mean any agency or entity of the government of the United States. If Licensee is a U.S. Government End User, then this Subsection (a) shall apply. The Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire the Software with only those rights set forth herein. The Software (including related documentation) is provided to U.S. Government End Users: (i) only as a commercial end item; and (ii) only pursuant to this Agreement.

12. Export Laws. Licensee shall not export, disclose or distribute any Software in violation of any applicable laws or regulations, including the export laws and regulations of the United States.

13. Severability. If any provision in this Agreement is invalid or unenforceable or contrary to applicable law, such provision shall be construed, limited, or altered, as necessary, to eliminate the invalidity or unenforceability or the conflict with applicable law, and all other provisions of this Agreement shall remain in effect.

14. Governing Law. This Agreement shall be governed by the laws of the State of California and the United States of America without giving effect to conflict or choice of law principles. The United Nations Convention on Contracts for the International Sale of Goods shall not be applicable and is rejected by the parties. The State and Federal Courts located in San Francisco County, California, shall have sole jurisdiction over any disputes arising under this Agreement and the parties hereby submit to the jurisdiction of such courts.

15. Entire Agreement. This Agreement is the entire agreement between the parties regarding this subject matter and supersedes all prior discussions, negotiations, agreements, and the like. This Agreement may be modified only in writing signed by authorized representatives of both parties. No vendor, distributor, dealer, reseller, sales person or other person is authorized by Zinc to modify this Agreement or to make any warranty, representation or promise which is different than, or in addition to, the warranties, representations or promises of this Agreement. Any purported oral modification of this Agreement shall be void.