

# INIFileTool-5M

## INI File Toolkit

**Version 1**

for Visual Studio .NET



***Desaware, Inc.***

Rev 1.0.2 (06/03)

## **Desaware, Inc. Software License**

Please read this agreement. If you do not agree to the terms of this license, promptly return the product and all accompanying items to the place from which you obtained them.

This software is protected by United States copyright laws and international treaty provisions.

This program will be licensed to you for use only on a single computer. If you wish to install it on additional computers, you must purchase additional software licenses. You may (and should) make archival copies of the software for backup purposes.

You may not make copies of this software for other people. Companies or schools interested in multiple copy licenses or site licenses should contact Desaware, Inc. directly at (408) 377-4770.

You have a royalty-free right to incorporate any of the sample code provided into your own applications with the stipulation that you agree that Desaware, Inc. has no warranty, obligation or liability, real or implied, for its performance.

Licensing: of the INIFileTool-5M uses the Desaware Licensing System Component. This framework provides for the transfer of licensing information from the system upon which INIFileTool-5M is installed, to Desaware's Licensing Web Service. This in turn creates and activates a license key that allows you to use the INIFileTool-5M components on your system. The licensing information transferred is a one way cryptographic hash that does not include any personal information, or information that could be used to identify the originating system. If you perform online registration, the registration information will also be transferred.

File Descriptions: You may distribute the file Desaware.INIFileTool.dll with your compiled assemblies. No other files may be redistributed.

Source Files: If you have purchased a source code license, the following applies: You may rebuild modified versions of the software provided subject to the restrictions listed. You may not use this source code to develop or distribute components and tools that provide functionality similar to all or part of the functionality provided by the INIFileTool-5M except for use with your own applications. Modified assemblies and namespaces must be renamed – you may not use Desaware in the assembly name or any namespace. However Desaware's copyright notice must be prominently displayed in any location where your own copyright notice is present. Source code may not be published or distributed, and may be used or accessed only by the individuals and at the locations covered by the source code license. You may distribute modified versions of the Desaware.INIFileTool.Dll file only for use by your applications. Modified versions of other components may only be used by the individuals and at the locations covered by the source code license.

### Limited Warranty

Desaware, Inc. warrants the physical CD and physical documentation enclosed herein to be free of defects in materials and workmanship for a period of sixty days from the date of purchase.

The entire and exclusive liability and remedy for breach of this Limited Warranty shall be limited to replacement of defective CD(s) or documentation and shall not include or extend to any claim for or right to recover any other damages, including but not limited to, loss of profit, data or use of the software, or special, incidental or consequential damages or other similar claims, even if Desaware, Inc. has been specifically advised of the possibility of such damages. In no event will Desaware, Inc.'s liability for any damages to you or any other person ever exceed the suggested list price or actual price paid for the license to use the software, regardless of any form of the claim.

DESWARE, INC. SPECIFICALLY DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Specifically, Desaware, Inc. makes no representation or warranty that the software is fit for any particular purpose and any implied warranty of merchantability is limited to the sixty-day duration of the Limited Warranty covering the physical CD and documentation only (not the software) and is otherwise expressly and specifically disclaimed.

This limited warranty gives you specific legal rights. You may have others, which vary from state to state.

This License and Limited Warranty shall be construed, interpreted and governed by the laws of the State of California, and any action hereunder shall be brought only in California. If any provision is found void, invalid or unenforceable it will not affect the validity of the balance of this License and Limited Warranty, which shall remain valid and enforceable according to its terms.

**RESTRICTED RIGHTS:** Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of Commercial Computer Software - Restricted Rights at 48 CFR 52.227-19, as applicable. Contractor/Manufacturer is Desaware, Inc., 1100 East Hamilton Avenue, Suite 4, Campbell,

Microsoft is a registered trademark of Microsoft Corporation. Visual Basic, Visual Studio, Windows, Windows 95, Windows 98, Windows ME, Windows NT, Windows 2000, and Windows XP are trademarks of Microsoft Corporation. INIFileTool-5M, FiveMinuteSoftware, Desaware Licensing System, CAS/Tester, SpyWorks, NT Service Toolkit, StateCoder, VersionStamper, StorageTools, Event Log Toolkit, ActiveX Gallimaufry, Custom Control Factory, and SpyNotes #2, The Common Dialog Toolkit are trademarks of Desaware, Inc.

Information in this document is subject to change without notice and does not represent a commitment on the part of Desaware, Inc. The software described in this document is furnished under a license agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Desaware, Inc.

Copyright © 2003 by Desaware, Inc. All rights reserved. Printed in the U.S.A.

# Table of Contents

<b>TABLE OF CONTENTS</b>	<b>4</b>
<b>INIFILETOOL-5M – FIVE MINUTE INI FILE TOOLKIT</b>	<b>6</b>
<b>Introduction</b>	<b>6</b>
<b>Features of INIFileTool-5M</b>	<b>6</b>
<b>Learning To Use INIFileTool-5M</b>	<b>6</b>
<b>Installation</b>	<b>6</b>
Samples	7
Licensing	7
<b>Deployment</b>	<b>9</b>
<b>Reference</b>	<b>9</b>
IniAccess	9
IniSection	10
IniException	11
<b>SECURITY</b>	<b>11</b>
<b>LICENSING</b>	<b>12</b>
<b>CUSTOMER SUPPORT</b>	<b>13</b>
<b>OTHER SOURCES OF INFORMATION</b>	<b>14</b>
<b><a href="http://www.desaware.com">www.desaware.com</a></b>	<b>14</b>
<b>Printed books by Dan Appleman</b>	<b>14</b>
Dan Appleman's Visual Basic Programmer's Guide To The Win32 API	14
Dan Appleman's Developing COM/ActiveX Components with Visual Basic 6.0: A Guide to the Perplexed	14
Moving to VB.Net: Strategies, Concepts and Code	14
Dan Appleman's Win32 API Puzzle Book & Tutorial for Visual Basic Programmers	15
<b>Ebooks by Dan Appleman</b>	<b>15</b>
Hijacking .NET: Volume 1: Role Based Security	15
Exploring .NET - Volume 1 - A collection of seven articles on .NET	15
Tracing and Logging in .NET	15
Obfuscating .NET: Protecting your code from Praying Eyes	15

Regular Expressions with .NET	15
Visual Basic.NET or C#? Which to Choose?	15
Introduction to NT/2000 Security Programming with Visual Basic	16
<b>Windows API Online Help</b>	<b>16</b>
<b>Microsoft's Developers Network CD Rom</b>	<b>16</b>
<b>Microsoft's Windows Software Development Kit &amp; Win32 Software Development Kit</b>	<b>16</b>
<b>DESAWARE PRODUCT DESCRIPTIONS</b>	<b>17</b>
<b>SPYWORKS PROFESSIONAL 7.0</b>	<b>17</b>
<b>DESAWARE LICENSING SYSTEM</b>	<b>18</b>
<b>STATECODER 1.0</b>	<b>18</b>
<b>CAS/TESTER – AUTOMATED CODE ACCESS SECURITY TESTING</b>	<b>18</b>
<b>VERSIONSTAMPER 6.5</b>	<b>18</b>
<b>NT SERVICE TOOLKIT 2.0 COM EDITION, .NET EDITION</b>	<b>19</b>
<b>DESAWARE EVENT LOG TOOLKIT</b>	<b>19</b>
<b>STORAGETOOLS VER 3.0</b>	<b>20</b>
<b>DESAWARE ACTIVE X GALLIMAUFY VER. 2</b>	<b>20</b>

# INIFileTool-5M – Five Minute INI File Toolkit

## ***Introduction***

INI files, or initialization files, are a simple text based file format that is often used to configure applications. The file is divided into sections, and each section is marked with the section name in brackets. Each section can have any number of entries, where the entry key is separated from its value using an equal sign. Comments are preceded by semicolons. A sample INI file might appear as follows:

```
; Comment for first section
[Section1]
First Entry=Some value
Second Entry=Another value

[Second Section]
; Comment for KeyName
KeyName = First entry value
```

INI files are not as powerful or flexible as the newly popular XML file format, but they are much easier for non-programmers to edit, usually using the Windows Notepad program.

## ***Features of INIFileTool-5M***

- Does not require unmanaged code permission needed by other approaches.
- Caching for maximum performance.
- Detection of INI file modifications by other processes.

As with all software in Desaware's Five Minute software line, the INIFileTool-5M component is easy to learn, use and deploy. Many developers will find the base component sufficient for most tasks, however a source code license is available for those who need additional customization.

## ***Learning To Use INIFileTool-5M***

The Desaware.IniFileTool.dll component contains two classes in the Desaware.IniFileTool namespace - IniAccess and IniSection. Add a reference to the Desaware.IniFileTool.dll component in your project to access these classes.

## ***Installation***

Run the INIFileTool.exe installation file and follow the prompts to install INIFileTool-5M on to your system. Two separate Windows Installer installations are included. One for installing INIFileTool-5M for Visual Studio .NET 1.0 and a second for Visual Studio

.NET 1.1. You should select the edition corresponding to the .NET edition you have installed. The rest of this document will refer to both of these editions as the same entity.

## Samples

The INITests sample application demonstrates most of the features of the INIFileTool-5M component, and includes VB.NET and C# source code.

## Licensing

The installation program will prompt you to enter your license key. Once you install the product, the installer will create a license file in the main product directory. This license file has a .DLSC extension.

The INIFileTool-5M component requires a license for each developer.

### INIFileTool-5M Quick Start

The following is an example depicting the use of the IniFileTool-5M component to read the contents of an INI file and retrieve the **Value** of the **UserName Entry** from the Registration Section.

[VB]

```
Dim objINIAccess As IniAccess
Dim objINISection As IniSection
Dim entryvalue As String

objINIAccess = New IniAccess(INI_file_name)
objINISection = objINIAccess.Section("Registration")
entryvalue = objINISection.Entry("UserName")
```

[C#]

```
IniAccess objINIAccess;
IniSection objINISection;
string entryvalue;

objINIAccess = new IniAccess(INI_file_name);
objINISection = objINIAccess.get_Section("Registration");
entryvalue = objINISection.get_Entry("UserName");
```

The following is an example depicting the use of the IniFileTool-5M component to read the contents of an INI file and set the **Value** of the **UserName Entry** from the Registration Section (creating the section and entry if they do not exist) and writing the changes back to the INI file.

[VB]

```
Dim objINIAccess As IniAccess
```

```

Dim objINISection As IniSection

objINIAccess = New IniAccess(INI_file_name)
objINISection = objINIAccess.Section("Registration")
objINISection.Entry("UserName") = "Willie the Kid"
objINIAccess.Write()

```

[C#]

```

IniAccess objINIAccess;
IniSection objINISection;

objINIAccess = new IniAccess(INI_file_name);
objINISection = objINIAccess.get_Section("Registration");
objINISection.set_Entry("UserName", "Willie the Kid");
objINIAccess.Write();

```

The following is an example depicting the use of the IniFileTool-5M component to read the contents of an INI file and verify that it contains a section named “**Registration**”. If not, the IniFileTool-5M component creates a new section with that name and adds new entries and entry values to that section. Finally, it writes the changes back to the INI file.

[VB]

```

Dim objINIAccess As IniAccess
Dim objINISection As IniSection
Dim sectionname As String
Dim sectionexists As Boolean = False

objINIAccess = New IniAccess(INI_file_name)

' check to see if section already exists
For Each sectionname In objINIAccess.SectionNames
    If String.Compare(sectionname, "Registration", True,
Globalization.CultureInfo.CurrentCulture) = 0 Then
        sectionexists = True
        Exit For
    End If
Next

If Not sectionexists Then
    objINISection = objINIAccess.Section("Registration")
    objINISection.Entry("UserName") = "Willie the Kid"
    objINISection.Entry("License") = "Enterprise"
    objINIAccess.Write()
End If

```

[C#]

```
IniAccess objINIAccess;
IniSection objINISection;
bool sectionexists = false;

objINIAccess = new IniAccess(INI_file_name);

// check to see if section already exists
foreach (string sectionname in objINIAccess.SectionNames)
{
    if (String.Compare(sectionname, "Registration", true,
System.Globalization.CultureInfo.CurrentCulture) == 0)
    {
        sectionexists = true;
        break;
    }
}

if (!sectionexists)
{
    objINISection = objINIAccess.get_Section("Registration");
    objINISection.set_Entry("UserName", "Willie the Kid");
    objINISection.set_Entry("License", "Enterprise");
    objINIAccess.Write();
}
```

## ***Deployment***

This component is XCopy deployable. Simply copy the Desaware.IniFileTool.dll file into the executable directory of your application or web application. No further configuration is required.

## ***Reference***

The following is detailed reference information for the Desaware INIFileTool-5Mclasses. Properties and methods of base classes are not listed unless they are overridden.

**Note on thread safety:** Unless otherwise noted, as is common in the .NET Framework, static methods of objects are thread safe, instance objects are not.

## ***IniAccess***

This class is used to read and write INI files using the following methods and properties:

Constructor	<p>VB: Sub New(ByVal filename As String)</p> <p>C#: IniAccess(string filename)</p> <p><i>filename</i> is the name of the INI file to edit. If the file does not exist it will be created.</p> <p>Exceptions: IniExceptionTypes - OnlyPrivateAllowed, MustHaveSection, InvalidEntryFormat.</p>
Write	<p>VB: Sub Write()</p> <p>C#: void Write()</p> <p>Writes the contents of the IniAccess object into the INI file specified in the constructor.</p> <p>Exceptions: IniExceptionTypes - ExternalModification</p>
SectionNames	<p>VB: Property SectionNames() As String()</p> <p>C#: string[] SectionNames</p> <p>Retrieves the section names of all the sections in the INI file.</p>
Section	<p>VB: Property Section(ByVal name As String) As INISection</p> <p>C#: INISection Section(string name)</p> <p>Allows you to set or retrieve the <i>IniSection</i> object. <i>name</i> is the section name of the section to set or retrieve and is not case sensitive. If <i>name</i> does not exist, a new section with the specified name will be created.</p>
DeleteSection	<p>VB: Sub DeleteSection(ByVal name As String)</p> <p>C#: void DeleteSection(string name)</p> <p>Deletes the specified section.</p>

## IniSection

This class is used to hold the entries for a given section using the following methods and properties:

DeleteEntry	<p>VB: Sub DeleteEntry(ByVal entryname As String)</p> <p>C#: void DeleteEntry(string entryname)</p> <p>Deletes the specified entry.</p>
Comments	<p>VB: Property Comments() As String()</p> <p>C#: string[] Comments</p> <p>Allows you to set or retrieve the comments for the section. This is Nothing/null if there are no comments for this section.</p>
Entries	<p>VB: Property Entries() As String()</p> <p>C#: string[] Entries</p> <p>Retrieves the entry names of all the entries in the section. This is set to Nothing/null if there are no entries in this section.</p>
EntryComments	<p>VB: Property EntryComments(ByVal entryname As String) As String()</p>

	<p>C#: string[] EntryComments(string entryname)</p> <p>Allows you to set or retrieve the comments for an entry. <i>entryname</i> is the name of the entry to set or retrieve and is not case sensitive. If an entry with the specified <i>entryname</i> does not exist, a new entry with the specified <i>entryname</i> will be created.</p> <p>Exceptions: IniExceptionTypes – InvalidKeyFormat.</p>
Entry	<p>VB: Property Entry(ByVal entryname As String) As String</p> <p>C#: string Entry(string entryname)</p> <p>Allows you to set or retrieve the value of an entry. <i>entryname</i> is the name of the entry to set or retrieve and is not case sensitive. If an entry with the specified <i>entryname</i> does not exist, a new entry with the specified <i>entryname</i> will be created.</p> <p>Exceptions: IniExceptionTypes – ValueTooLong, InvalidKeyFormat.</p>
Name	<p>VB: Property Name() As String</p> <p>C#: string Name</p> <p>Allows you to set or retrieve the name of the section.</p> <p>Exceptions: IniExceptionTypes – InvalidSectionFormat.</p>

Note that the IniSection class does not include a constructor. You can create a new instance of this class using the IniAccess class's Section property.

## IniException

This exception class is used to return exception information for the INIFileTool-5Mcomponent:

Enum IniExceptionTypes	<b>OnlyPrivateAllowed</b>	For security reasons, this class does not allow access to win.ini or system.ini.
	<b>InvalidSectionFormat</b>	The section name is not valid.
	<b>MustHaveSection</b>	No section found in the INI file.
	<b>InvalidEntryFormat</b>	The entry has an invalid format.
	<b>ExternalModification</b>	INI file has been modified externally.
	<b>ValueTooLong</b>	Value exceeds the maximum length.
	<b>InvalidKeyFormat</b>	The entry (key) name is not valid.
	<b>Undefined</b>	Undefined error.

## Security

The INIFileTool-5M component is implemented in managed code. INIFileTool-5M does no asserts. It does support partial trust components, but those components must have the

necessary permissions to access the INI file. However, it does not require unmanaged code permissions.

## Licensing

The Desaware IniFileTool-5M component is available in two licensing options.

- The Standard license requires a separate license and installation code for each developer seat. There are no royalties for distributing the component with your assembly.
- The Source license includes a site license for all developers in a single geographical location (one street address). You may modify the source code and distribute modified components for use in your own applications subject to the limitations that follow.
  - ◆ You may not sell, market or re-license modified components, or components or tools that incorporate our source code, for use by other developers to develop their own INI file tool. In other words – you can't modify our code, then sell a product that competes with this toolkit in any way. You can only use modified versions of our code to facilitate the development of your own, non competing, products.
  - ◆ If you modify our components, you must change the assembly name and all namespaces so that they do not include the word Desaware.
  - ◆ Your assembly.vb file must be modified to your own product name.
  - ◆ Your assembly.vb file must, as part of its copyright notice, include “Portions of this code are copyright ©2003 by Desaware Inc.”

## Customer Support

We at Desaware have one very simple company policy - we do our best to treat our customers as we would like to be treated (after all, we are programmers too).

If you have purchased this software directly from Desaware and you feel that this program is not for you or you are otherwise dissatisfied, please feel free to return it for a full refund (if you purchased it elsewhere you will need to contact your dealer for return or refund information - also, we reserve the right to limit this offer to 30 days from the invoice date). Your satisfaction is important to us, and we are well aware that this is a very unusual product and not appropriate for everyone.

Please address all correspondence to:

Desaware Inc.  
1100 East Hamilton Avenue, Suite 4  
Campbell, CA 95008

Phone (408)377-4770, Fax (408) 371-3530

Internet: <http://www.desaware.com>, <http://5minutesoftware.com> or  
[support@desaware.com](mailto:support@desaware.com)

## **Other Sources of Information**

Here are several other resources that we recommend for advanced Windows development.

### ***www.desaware.com***

Desaware's web site includes numerous technical articles on all aspects of Windows development. Be sure to also check the FAQ and support section for this product.

### ***Printed books by Dan Appleman***

#### **Dan Appleman's Visual Basic Programmer's Guide To The Win32 API**

Written by Daniel Appleman (president of Desaware) and published by McMillan, (ISBN 0-672-31590-4) - this sequel to the original 16 bit API Guide applies the same philosophy to teaching the Win32 API to developers using Visual Basic and VBA based applications. With more examples, more functions, more tutorial style explanations and a full text searchable electronic edition on CD-ROM, this book should prove a worthy successor to the 16 bit API book. Covers Visual Basic version 4 through 6.

Available at most good bookstores, or directly from Desaware at a 20% discount - call (408) 377-4770 or email [support@desaware.com](mailto:support@desaware.com).

An upgrade CD is available for owners of the "PC Magazine's Visual Basic Programmer's Guide to the Win32 API" ISBN: 1-56276-287-7 for \$24.99 + s&h directly from Desaware. Refer to our web site at [www.desaware.com](http://www.desaware.com) for additional information.

#### **Dan Appleman's Developing COM/ActiveX Components with Visual Basic 6.0: A Guide to the Perplexed**

Written by Daniel Appleman (president of Desaware) and published by McMillan, (ISBN 1-56276-576-0) - this book is designed for those programmers interested in using Visual Basic's object oriented technology to develop ActiveX components including EXE and DLL servers, ActiveX controls and ActiveX documents. Unlike many books that simply rehash the Visual Basic documentation, this one serves as a commentary to clarify and extend the documentation. Of special interest to VersionStamper customers will be the chapters on OLE and COM technology that will help them further understand the process of registering components, and the chapters on versioning and licensing.

The VB6 version also includes two new chapters on IIS Application development.

Available at most good bookstores, or directly from Desaware at a 20% discount - call (408) 377-4770 or email [support@desaware.com](mailto:support@desaware.com).

#### **Moving to VB.Net: Strategies, Concepts and Code**

Written by Daniel Appleman (president of Desaware) and published by Apress (ISBN# 1-893115-97-6).

VB.Net is not Visual Basic. COM+2.0 is not COM.

Porting is stupid. These are just a few of the things you'll learn as Dan takes you on a journey unlike any other into the world of VB.Net. He tackles strategic issues to help you determine when and whether to deploy VB.Net.

As always, Dan teaches the core concepts such as inheritance and multithreading, where VB6 programming habits can lead to costly design and development errors. And he covers the language changes to help you adapt to, and understand, this new environment.

## **Dan Appleman's Win32 API Puzzle Book & Tutorial for Visual Basic Programmers**

Written by Daniel Appleman (president of Desaware) and published by Apress (ISBN# 1-893115-01-1). Appleman's Win32 API Guide covers 700 API functions. This book covers the other 7800. How? By teaching you everything you need to know to read and understand the Microsoft C documentation and create correct API declarations for use in Visual Basic. Presented in an entertaining puzzle/solution format that challenges you to solve real world API problems. In depth tutorials take you behind the scenes to understand what really happens when you call an API function from VB.

### ***Ebooks by Dan Appleman***

#### **Hijacking .NET:Volume 1: Role Based Security**

You'll learn a few lines of .NET code can allow you to determine which roles (groups) an account belongs to. How to extend several .NET classes using hidden method calls and a few API calls to allow you to easily set the account and group security for a file or directory.

#### **Exploring .NET - Volume 1 - A collection of seven articles on .NET**

Covering subjects from control arrays, to the internals of .NET strings, to creative uses of attributes and more!

#### **Tracing and Logging in .NET**

An in-depth tutorial on .NET's tracing diagnostic system.

#### **Obfuscating .NET: Protecting your code from Prying Eyes**

Details how easy it is to decompile .NET applications. Includes a license to down-load and use Desaware's open source QND- Obfuscator.

#### **Regular Expressions with .NET**

An in depth tutorial on the rarely mentioned "fourth language" in Visual Studio .NET. With its powerful text processing and input validation capabilities, this e-book is essential reading for every .NET developer.

#### **Visual Basic.NET or C#? Which to Choose?**

This controversial e-book by Dan Appleman directly challenges the common wisdom as to which language is really best.

## **Introduction to NT/2000 Security Programming with Visual Basic**

NT Security is a subject that is intimidating, to say the least. This article, gives developers the foundation of knowledge needed to understand even the most obscure security concepts.

### ***Windows API Online Help***

The Professional Edition of Visual Basic includes Win31api.hlp and/or win32api.hlp - an online help reference for all API functions. These functions are declared in C and do not consider Visual Basic compatibility issues, however the information in chapter 3 of the Visual Basic Programmer's Guide to the Windows API (chapters 3 and 4 of the 32 bit book) will provide you with information on how to translate these functions to Visual Basic.

### ***Microsoft's Developers Network CD Rom***

This amazing CD-ROM and web site (<http://msdn.microsoft.com>) contain a wealth of information and sample code, plus the latest Visual Basic knowledge base.

### ***Microsoft's Windows Software Development Kit & Win32 Software Development Kit***

The sample code is all in C, but by the time you've read the Visual Basic Programmer's Guide to the Windows API or Win32 API, you'll know enough to be able to translate the C code to Visual Basic.

## Desaware Product Descriptions

Thank you for your purchase of this Desaware (or 5 Minute Software) product. We have additional quality software to enhance your programming efforts. Please visit our web site at [www.desaware.com](http://www.desaware.com) (or [www.5minutesoftware.com](http://www.5minutesoftware.com)) for detailed descriptions and product demos.

### **SPYWORKS PROFESSIONAL 7.0**

#### **SpyWorks in a nutshell? Impossible!**

You're going to want to download the SpyWorks demo to even begin to understand its capabilities. This product has been evolving for several years, and it includes so many features it's hard to know where to begin. SpyWorks is a VB power tool. When you need to override VB's default behavior or to extend VB's functionality, you will want to use SpyWorks.

#### **Do That in Visual Basic??**

Want to put VB to the test? Want to learn advanced programming techniques? Want to keep the productivity of VB and have the functionality of C++? SpyWorks contains the low level tools that you need to take full advantage of Windows. Here are just a few of the features of this multi-faceted software package. For instance, have you ever wanted to detect keystrokes on a system-wide basis or detect when an event occurs in another application or thread using subclassing or hooks? SpyWorks can help you solve these problems by letting you tap into the full power of the Windows API without having to be an expert. SpyWorks lets you export functions from VB DLL's so that you can create function libraries, control panel applets, and NT Services. With its ActiveX extension technology, you can call and implement interfaces that VB5 or 6 do not support.

SpyWorks includes the Desaware API Class Library, which assists programmers in taking advantage of the hundreds of functions that are built into the Windows API.

The Professional Edition includes .NET support for keyboard hooks, window hooks and subclassing (including cross-task subclassing) with examples in both Visual Basic.NET and C#. Additionally, a WinSock component with comprehensive VB source code that gives you complete control for Internet/intranet programming.

Other features are the NT Service Toolkit *Light Edition*. This application is a subset of the Desaware NT Service Toolkit product. It allows a developer to create true NT services using Visual Basic. A background thread component that allows you to easily create objects that run in a separate background thread.

It also contains extensive sample code.

The Professional Edition includes the Winsock Library, NT Service support and many other additional features. SpyWorks 2.1 (VBX Edition) is included in the Pro Edition.

**ActiveX/COM/.NET, VB 4-6, Visual Studio .NET, Windows 95/98/ME, Windows NT/2000/XP**

## ***DESAWARE LICENSING SYSTEM***

- ◆ For July 2003, Desaware's Licensing System is easy to use, easy to learn and affordably priced. No additional fees or commissions, and no additional charges for protecting more than one program.
- ◆ End-to-end cryptographic based, avoids the need for “secrets”, and makes it possible to offer an optional full source code license to the technology.
- ◆ Cryptographic certificate licensing avoids the need to modify the client system. No hidden files, registry entries, or background services needed.
- ◆ Supports licensing of partial-trust components once the licensing component is installed on a client system. Ideal for “no-touch” deployment and web deployment scenarios.
- ◆ Allows a variety of security levels from high (activation required), to low (activation optional) allowing it to be used even on systems without full time Internet connections.

## ***STATECODER 1.0***

A .NET class framework that makes it easy to create and support powerful state machines using VB .NET or C#. Dramatically improves the reliability of applications, components and services that make use of the multithreading and asynchronous features of .NET.

**Visual Studio .NET, Windows NT/2000/ XP**

## ***CAS/TESTER – AUTOMATED CODE ACCESS SECURITY TESTING***

How will your .NET assembly work if someone suddenly disabled file access? Registry access? Its user interface? Desaware's CAS/Tester automates the process of testing an assembly (for Code access security) by executing it under multiple security configurations and producing a detailed report of the results. Features include: more than 80 tests included, or create your own using VB .NET or C#; supports testing of class libraries, components, user controls, windows forms applications and console applications. **Visual Studio .NET, Windows NT/2000/ XP**

## ***VERSIONSTAMPER 6.5***

### **Distributing Component-Based Applications? Beware DLL HELL!**

You've distributed your application and it's working fine. But your end user is still in charge of their system. What happens when they install a program that overwrites a component that your software needs to run? Can you verify that your users have the correct files required by your application? Can you really afford to spend two hours on the phone trying to figure out exactly what went wrong? Now you can easily avoid component incompatibilities by adding VersionStamper to your toolkit. It lets you check the versions of your program's components on your end user's system, and correct the problem.

**You are in control!**

DLL Hell is a big problem, and with VersionStamper you can be in control of how this problem is detected and corrected. You determine dependency scanning (file size, date, version or other parameter), how and when the dependency scanning is done (upon start up, at midnight, at user's discretion), and how you want the problem resolved (automatically, an email message to your help desk, from a dependency list on your web site and more). This means you can handle versioning problems as simply as using a message box to call tech support, or even automatically updating the invalid components over the internet or corporate network. Imagine your application updating itself without user (or programmer) intervention! Imagine the hours and money saved in tech support calls! You can even use VersionStamper for incremental updates and bug fixes.

### **Is This For Real?**

No, you don't have to pay a fortune in distribution fees - there are no run-time licensing fees. VersionStamper comes with a great deal of sample code. Don't distribute a component-based application without it!

Checks the versions of your dependent files and notifies you or the user of potential problems.

Internet extensions allow you to update versions across the Internet/intranets.

Cool and USEFUL sample programs show you how it works.

Includes VB source code for the VersionStamper components that you can use in your applications. **ActiveX/COM, VB 4-6, C++ 5/6, Windows 95/98/ME, Windows NT/2000**

### ***NT SERVICE TOOLKIT 2.0 COM EDITION, .NET EDITION***

Create a fully featured service in minutes using Visual Basic – even debug your service using the Visual Basic environment! Supports all NT service options and controls. Adheres to all Visual Basic threading rules. Background thread support allows easy waiting on system and synchronization objects. Client requests supported on independent threads for excellent scalability, with client impersonation available allowing services to act on behalf of clients in their own security context. Client requests and service control possible via COM/COM+/DCOM.

Simulation mode for testing as an independent executable. Create control panel applets for service control and other purposes.

**COM Edition: COM, VB 6.0, Windows NT/2000/XP**

**.NET Edition: Visual Studio .NET, Windows NT/2000/XP**

### ***DESAWARE EVENT LOG TOOLKIT***

Visual Basic allows you to log events to the NT/2000 event log, but does not allow you to create custom event sources - so every event belongs to the application VB runtime, descriptions are limited, and event categories unavailable. Even if you use the API to log events, creating custom event sources for your application is not supported by VB, and is difficult with C++.

Desaware's Event Log Toolkit makes creation of event sources easy, and provides all the tools needed to create and log custom events. Now your applications and services can

support event logs in a professional manner, as recommended by Microsoft.  
**ActiveX/COM, VB 5/6, Windows NT 4/2000**

### ***STORAGETOOLS VER 3.0***

StorageTools is your key to the OLE 2.0 Structured Storage Technology. Structured Storage allows you to create files that organize complex data easily in a hierarchical system. It is like having an entire file system in each file. OLE 2.0 takes care of allocating and freeing space within a file, so just as you need not concern yourself with the physical placement of files on disk, you can also disregard the actual location of data in the file.

Additionally, with its support for transactioning you can easily implement undo operations and incremental saves in your application.

StorageTools allows you to take advantage of the same file storage system used by Microsoft's own applications. In fact, we include programs (with Visual Basic source code) that let you examine the structure of any OLE 2.0 based file so that you can see exactly how they do it!

StorageTools includes registration database controls for Windows NT, Windows 2000/XP, Windows 95 & 98. Plus, a simple resource compiler (with source) so that you can create your own .RES files for use with Visual Basic and more. 16 & 32 bit COM/ActiveX and .NET.

***New for version 3.0!*** StorageTools 3.0 includes .NET support for accessing OLE Structure Storage from .NET assemblies. **ActiveX/COM/.NET, VB 4-6, Visual Studio .NET, Windows 95/98/ME, Windows NT/2000/XP**

### ***DESAWARE ACTIVEX GALLIMAUFREY VER. 2***

#### **What is it?**

gal·li·mau·fry (gàl'e-mô'frê) noun

plural gal·li·mau·fries

A jumble; a hodgepodge.

[French galimafrée, from Old French galimafree, sauce, ragout : probably galer, to make merry. See GALLANT + mafrer, to gorge oneself (from Middle Dutch moffelen, to open one's mouth wide, of imitative origin).]

(From The American Heritage® Dictionary of the English Language, Third Edition copyright © 1992 by Houghton Mifflin Company)

What does a Twain control, spiral art program, set of linked list classes, a quick sort routine, a hex editor and a myriad of other custom controls have in common?

They are all part of Desaware's ActiveX Gallimaufrey.

You'll find most of these controls useful, the rest entertaining – but we guarantee that you'll find them all educational, because they come with complete Visual Basic 6.0 source code.

### **Curious?**

Want to learn some advanced API programming techniques? Visit our web site for a full description and demo. **ActiveX/COM, VB 4-6, Windows 95/98/ME, Windows NT/2000**

### ***Custom Control Factory Ver. 4.0***

The Custom Control Factory is a powerful tool for creating your own animated buttons, multiple state buttons, toolbars and enhanced button style controls in Visual Basic and other OLE control clients, without programming. With 256 & 24 bit color support, automatic 3D backgrounds, image compression, over 50 sample controls and more. Plus MList2 - an enhanced listbox control. 16 & 32 bit ActiveX controls and 16 bit VBXs included.