AHM Component Collection VEnhanced97

The Beta Release 3 for Delphi 3.0
and the Final for Delphi 2.0

Freeware until 11.October 1997

written by

Alexander H. Mehlhorn

This Collection has obtained the **Delphi Choice** *Award in May 1996*

Please read the following sections about the components, if this help isn't enough contact me at ahm@paradigm-sa.com

The Components

The Tools

What's new and what is old
About me and the Collection
How to install and Trouble Shooting
About older Releases of the Collection
Ordering Information

My new URL is http://www.paradigm-sa.com/~ahm

A big THANK YOU to all the people who have registered up to today and helped me maintain this Collection !

Smile !   It makes no Enemies and Life easier !

- Alexander Mehlhorn -

This Help file was developed using Microsoft Word together with the Microsoft Help Workshop
Delphi is a Trademark of Borland International

AHM Component Collection VEnhanced97

On the Palette

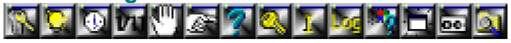
This version is considered an upgrade. Problems have been fixed in many components
and new properties were added where necessary, please keep in mind that that might change existing
Software developed by you !

AHM Controls/1

AHM Controls/2

AHM Data Controls

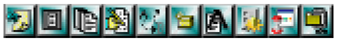
AHM Dialogs



AHM System/1

AHM System/2

AHM Files



AHM Component Collection VEnhanced97

About it all

Starting to develop in Delphi in the middle of '95 I have considered it to be the best tool on the market today. Research from various Companies has shown that Delphi outruns the Powerbuilders,VB and other Client/Server tools by far. This is a new time of development, its a new market and its a different point of view one has to take to properly develop Applications today. I have started to develop Components in the beginning of '96 trying to get you the best tools to develop your Application in Delphi. This Collection consists out of the most needed and graphically best Components I could develop over the previous months. After winning the Delphi Choice award in May '96 I have to thank all those who registered and thereby made it possible for me to develop this Toolbox to hopefully one of the best on the market today. If you have not seen this toolbox simply download it and install it. It should get you the best results for development. After using them watch those client of yours go " hey this is cool " when they run your Application. I have had confident feedback that these Components are helpful and hope that you will agree after downloading it.

About me

Well my name is Alexander Mehlhorn as you would have figured out by now and I am a 4GL developer at a firm called Paradigm in South Africa. Actually I am born in Germany/Leipzig, grew up there and came here about 5 Years ago. Further   I am busy with my studies at Unisa and in my private life I enjoy playing pool if   not just busy with developing. Well I hope that’s enough about me to cut it short ! If you want to know more contact me via e-mail. In case you are looking for a serious Delphi Developer E-Mail me and send me the Details otherwise find my **CV** attached in this Helpfile.

Now what have we got ?

Well by now these Components range from 3D Hints over Gradientbuttons to
3D InternetExplorerbuttons to scrolling Images and,and,and ....
Why so many ? Well I would like to supply you with the best Components to develop your Application in a professional way.
Have I used resources of other people you wonder ? Yes and No. I have integrated some Components after they have been modified by me. Those Components have some good functionality that I wanted to have in my Collection. However most Components are build by me on the basic Idea of the Component itself. I am in need of more good Components which you need for your Applications. So if you have any Ideas let me know. I usually make plans for creating them. A big thank you goes to Larry Yenta,Michael Wakkinen and to Thomas Speck (three registered Users of the Collection) who have given me a lot of Hints and Ideas on how to create better and new Components. TSerialNumber was an Idea of Michael and I have integrated it into the Collection. If you have more such good Ideas let me know.

Websites supporting the AHM Component Collection

Well first I need to mention Christine here. **Christine at SofTec Inc.**

- **URL : http://www.internetwantads.com/softec/comp.htm**

Not only has she given nice comments to the Components. She is also advertising all over her Webpages for my Collection and has put up a Bitmap displaying some of my Components. Thank you Christine for your continues support of the Collection. Hey and if you looking for cool guestbooks for your Website then go visit her.

Next I would need to mention the **Delphi Superpage.**

- **URL : http://sunsite.icm.edu.pl/delphi//**

Now as many of you might know the Delphi Superpage in Poland has been around for a long time now. The Delphi Superpage has not only supported me with the Collections but has also given me good Ideas of new Components and has been a site that is bookmarked by me for a long time now. In my opinion without the Delphi Superpage Delphi would not have the status it is in today.

Well and there is the site that keeps coming up and going down. From sickenesses over Bike accidents over Computer Problems it is always a great site to visit when David is supporting it. I hope he can sort out those troubles he has and make it work. Well the sites name is known to many of you. Its the **Delphi Information Connection**.

- **URL : http://www.delphi32.com/**

Thanks to David and his Team who have given me the Delphi Choice Award in May '96 I am also supported greatly by his nomination.

Well the next two sites are two sites I have found travelling the Internet. There has not been much advertisement for them yet they are becoming great resource sites for Delphi Components and I must say that over the past month many people have visit me after coming from either one of those two greatlooking websites. The Pages I am talking about belong to the
Delphi Deli and Torry's Delphi Pages

- **URL Deli : http://www.intermid.com/delphi/**
- **URL Torry : http://carbohyd.siobc.ras.ru/torry/**

Both sites have given me the ability to add my Components very quickly and have added some nice comments to it. If you are looking to release any Components yourself contact them and I promise you won't be disappointed. If your are looking for great resources, well use them aswell.

A smaller but not less useful resource I found in **The Temple of Delphi**. The Temple is another great site for resources and has always added my newest releases without any problems whatsoever. If you want to go and visit here is the URL :

- **URL : http://www.coast.net/~jkeller/**

Well and at last but not least here is one of the biggest Component Collectors out there. **The Delphi Exchange**. Well and its not only about Components. Here you get to know and maitain yourself anything you want to. This site has maintained for a long time each and everyone of my releases without my knowledge. I was informed a few weeks ago that it exists and I was shocked by the great way this site is implemented. It is a pleasure to work with Brad Choate the maintainer of this Site and I have not had any problems after my original jumpoff to the site. If you want to get a great link on resources for Delphi use the following :

- **URL : http://www.delphiexchange.com/**

Well I hope I have not forgotten anyone. There are however many more great sites out there. These are just a few of the ones that have supported me. I thank everyone again who has put up a link to the AHM Component Collection and I have to say that I would not have done without you.

Well here is another link. This link goes to my own Website for those of you who have not had a way to find out, I have a new and permanent Website maintained here. Its link is :

- **URL : http://www.paradigm-sa.com/~ahm**

If you come around to visit please fill in the guestbook and enjoy the trip around. I know I have problems with the pages from time to time. If you find one just mail me. Afterall I am not perfect myself.

The Helpfile …

Well here is the Helpfile and I hope that it helps everyone out there who is looking for some decent Documentation. Yet I have not integrated it with Delphi. Why not - you ask ? Well if I have a look at all those people writing helpfiles for integration with Delphi I think that the actual Database for the Help is becoming so large that it is unmaintainable for any Delphi User. If you want Help on these Components just click on the Icon for it and it will come up and take you to the Component you are looking for. Also if you want future releases to include somekind of examples on specific Components - please you welcome

to e-mail me and I will build it in. I am on my own out here and to build extra large helpfiles is not really my best. I will help you out whereever you require a little Help, yet try and see if the Help provided here is not sufficient enough for you.
This Helpfile however will be maintained for all future Releases and I hope to integrate it with an Install Program the next time I get around to it. I will however leave you the Options to download it as an individual File from my Website since the File is quiet large and I wouldn’t want it to become a hassle for you to download my collection.

Something else ?

No not really, just enjoy this package and try to find something out there that will replace its value. My registered users certainly haven't. Well and if you really want to make me happy in a small way, visit my Website at **http://www.paradigm-sa.com/~ahm** and enter my Guestbook.

Happy development and keep smiling, your Alexander Mehlhorn (AHM)

The AHM Tools

The AHM Tools is my newest Invention. Well not really. I needed a way to get the common Tools I use into Delphi. And some of them are here now. It comes by default with my Collection and will be maintained or increase. If you have special requirements let me know and I see what I can build in for your. Good Ideas I can always use. However I will not document the Tools nor do any type of support on them. I can not affoard it since I do not have the time nor money to do that. My main interest is with the components so to document the tools now aswell is not really my purpose. Also I believe that whoever will use the Tools is considered a Developer and I hope that this person does not need explanations on how INI-Files look or what the Idea of Process Termination is. One thing about them thought... I am building on whatever I consider to be a good Idea for the Tools and if you do have a problem contact me and I will see what I can do for you.

The Tools are totally Freeware and therefore belong to the public. Source Code will not be made available so don’t ask me about it. The only Source Code that is for Sale is the Source of the Components once you register and pay for it.

The Tools also display you a little of the things the Components can do for you and therefore will always require the Components to be installed. So if you don’t install the Components don’t ask for the Tools to be installed.

Delphi 2.0

I am sorry people ... Since I make use of Packages that is an Idea of Delphi 3.0 only I can not support it under Delphi 2.0. Time to move and get the most powerful Development Tool on the market today - Delphi 3.0.

AHM Component Collection VEnhanced97

How to install

- Go into Delphi 2.0
- Choose Component/Install Components from the menu
- Choose Add   Components

  Unregistered Version or Registered without Source :
  - Switch the Type to *.dcu and select AHMRegistration.dcu from directory

  Registered Version with Source :
  - Switch the Type to *.pas and select AHMRegistration.pas from directory

- Press Ok to compile

- Go into Delphi 3.0
- Choose Component/Install Packages from the menu
- Choose Add   Package

  - Select AHMD3 from directory

- Press Ok to compile

Troubleshooting

After the e-mail I received on troubles people had installing here is just a little help :
If you install collections like this one the chance is big the one or more components you previously added have similar names. Please remove all components that do not come with Delphi from your Comlib32.dcl (Library) then add these two files as indicated above !
Recompile and add the others back one by one and you will find the ones giving the problem you can then decide which one is easier or nicer for you to use !
Also for all those people who removed the original setupstrings from Delphi the aliases need to be properly defined.
e.g. Wintypes=Windows,Winprocs=Windows ...

Known Problems

I know about **Problems with TSpeakerSound and THints** under Windows NT with certain Windows Configurations. I am working to solve the Problem but since I threw Windows NT 4.0 out after its final release and the many,many bugs it still has I have not reinstalled/tested the Collection under NT. For any Problems whatsoever please contact me via E-mail and I will make a plan to fix them.
To all the people who have given me feedback on bugs I have a big thank you to say since you have also helped me to create a good and stable Collection of Components.

Secondly the biggest problem I currently have is that when you **going from one release to another** of the AHM Component Collection your Application finds problems with it. I found the only way to treat this Problem is to follow the following steps :

- have the old release installed
- open all the forms having the Components that give you problems
- write down the properties you have changed (compared to the default Component)
- write down the names of the events you have attached

- DO NOT delete the procedures of the events
- delete just the Component from the form
- Install the new release of the Collection
- Drop the same Components back onto your form
- reattach the events using the dropdown of the event
- reinsert the values for the properties
- DONE

Okay now let me explain why this happens. Properties of Components are the most likely to change in any new release of Components. There might be new Properties even. Old properties might be removed. Any of those possible reasons will find problems with the Form the Component is on when opening it since it tries to read the properties out of the .DFM file.
I am really sorry that this happens but Delphi has not inco-operated some method to handle the change of properties in Components (Example would be to skip any nonexisting property or use its default if its new). I have not kept previous Releases of the Collection when I went to AHM V97.32 beta 1 since I updated the registered Users automatically. I hope you wont find too many Problems with it. If so contact me and I will help you however possible, yet I can not promise that there will be much I can do about it. This is afterall a Component Collection and Properties do change again and again as functionality gets better.
I hope that this will help you a bit and wish you the best of luck with your development.

Zipfile unzips using short Filenames ?!?
Get a version of Winzip or similar. Something that can handle long filenames.

File is missing ?!?
Most likely you have unzipped the Archive using a 16-Bit Application. You files consist out of short filenames. If the problem still persists contact me and I will help you whereever possible.

About Collection 1.0 and 2.0 for Delphi 1.0

Well I will still send them out once you register but since it is not quiet useful to have Delphi 1.0 and 2.0 installed I will not support them any longer . If there are big problems with something and you are a registered user I will however support you . Thanks by the way to all the people who have registered these Components . I am looking for someone from Greenland/America who had registered with me and received the Components . I want to send you the new Version of them but I can not get your e-mail Address since I lost yours when I had a harddisk crash a little while ago . If you could just let me know again what it is or send some mail I would appreciate your understanding .

Other Releases

When I went to Version 97.32 and on to VEnhanced I changed nearly everything possible to new formats. Timers became threads, I created classes instead of just properties … etc. Because of that I just threw all old Versions away and the only one that is applicable to you should be V97.32 the final or this Beta. This final I will support even if new Releases come out. (So you could register for the next release and receive the registered Version of this one for nearly nothing). I has come to my attention that that might create Problems for you if you have installed e.g. V3.32 and you want to easy go over to V 97.32. ! Components change, new ones get added, old ones get removed, properties change. I am sorry that I have no other way than the one under Trouble Shooting to solve this Problem. Stay on top and register now. I will send you any new release the moment it comes out and you can make easy transitions to the new Version.

AHM Component Collection VEnhanced97

Ordering Information

Let me list you the packages :

- **$119 US for the full Component Package without Source Code**
- **$249 US for the full Component Package with Source Code.**

Note that is still less then two dollars per Component and I have not increased the price at all in my latest releases. Also I am currently not planning to. Keep supporting the Components and it won't increase even thought the Components, the Helpfile, the Website and the Releases are all controlled in my free time.

I have decreased the actual price due to the many Students out there wanting to buy the AHM Component Collection but not being able to afford the old prices.

Upgrading :

Going from one final release to another will cost you $60 without Source and $120 with Source, but since this is my first final release it will not cost anything's as such. If you a registered User of one release without Source and you now plan to purchase the Source add the difference between the actual Cost to the $120 to get the cost involved.

The way to do this is to send a International Money Order by mail to

Alexander Mehlhorn
POBox 11037
Verwoerdburgstad
Hennopsmeer
0046
South Africa

or e-mail me and I will give you my Bank Detail on which you can receive the Package within 4 Days (That's how long it takes if you pay directly into my account).
On any International Money Order I ask you to include $20 for Bankfees.
If you come from a different country than the USA please make sure the Money Order contains at least something equal to the amount indicated.
Now once I receive a Money Order you should also supply me with your
e-mail Address. This Address will be used by me to send out the registered Version straight to you ! This version will be supported which means whenever there are upgrades you will receive it and when you have problems they will be addressed as soon as possible. Further will you be supplied with any Information regarding new releases and other interesting things !

Freeware until 11.October 1997

This Software is Freeware until the 11.October 1997 after which it becomes Shareware. It will still work and have all the functionality. Yet it will show you Dialogs indicating which Component of the Release you are using from within your Application.

My e-mail is :
ahm@paradigm-sa.com

- **Even if I am specifying not to be selling the AHM Tools nor distributing the Source I will give you an Idea on what you can do if your Company (as I believe that only Companies would insist on this) can do to obtain the Components and the Tools with Source. The Price for**

obtaining the AHM Tools with Source + the AHM Components with Source would be $750 as I can not give you credit for all those people out there who are students and private people.

AHM Controls/1

- TBackground
- TCheckbox3D
- TRadioButton3D
- TMultiLabel
- TMultiBtn
- TIE3DButton
- TOffice97Btn
- TMultiPanel
- TDigitalClock
- TMultiGauge
- TMultiPresentation
- THints
- TFillBackground
- TArrowBtn
- TRoundBtn
- TColorDice
- TMultiHeader
- THeaderList

AHM Controls/2

- TColorBevel
- TColorCombo
- TFontCombo
- TWinStringGrid
- TWinCheckboxList
- TWinListBox
- TLed
- TDigital
- TLookupEdit
- TAnimator
- TImage3D
- TImagescroller
- TFadingLabel
- TOffice97Edit
- TScreenCanvas
- TClipboard

AHM Data Controls

- TDBSearchForm
- TWinDBListbox
- TDBLookupEdit
- TDBMultiLabel
- TAsciiDBTransfer
- TDBMultiNavigator
- TDBPasswordDlg
- TDBOffice97Edit
- TDBIndexLookup

AHM Dialogs

- TPasswordDlg

TTipTrickDlg
TTimePopup
TDatePopUp
TMsgBox
TMsgDlg
TAboutInformation
TAccess
TQuickInfo
TLogWindow
TShellAbout
TSplashScreen
TFolderBrowser
TSerialNumber

AHM System/1

TIniEntry
TRegEntry
TTimeTrigger
TTrialRun
TApplicationInfo
TDriveVolumeInfo
TQuickMail
TResourceInfo
TSystemInformation
TSystemTray
TWavePlayer
TSpeakerSound
TTimer32
TExitWindows
TMenuItemGlyph
TJokeHolder
TOwnerShip
TStringHolder
TSystemScreenSaver

AHM System/2

TOnlyOne
TFormExplode
TFormTopMost
TFormSizeOptions
TSystemMenu

AHM Files

- TFileOperator
- TTextFileView
- TFileTextReplace
- TFileCollector
- TDroppedFiles
- TMultiFileLabel
- TLogFile
- TFileProperties
- TFileExecute
- TZipFileView

TDroppedFiles

TDroppedFiles = Class(TComponent)

A Wrapper around Windows DragQueryFile. This Component will collect all Filesnames of Files dropped onto your Form. Try it with Explorer . Set the Component to Active and Drag some Files onto the Form. It even works in Design.

Properties & Events :

AboutComponent	Component Information
Active	Enables/Disables the receiving of dropped Files
Collect by	Type of filename to collect
DroppedCollection	Property that holds the collected Items dropped onto your Form

TSystemScreenSaver

TSystemScreenSaver = Class(TComponent)

A Wrapper around Windows ScreenSaver Functionality.

Properties & Events :

AboutComponent	Component Information
Active	Disable/Enable Windows ScreenSaver
DsgnExecute	Execute in Design Time
SecondsToActivate	Set the number of Seconds until the Screensaver automatically activates

TFileProperties

TFileProperties = Class(TComponent)

Read and set the properties of an existing File.

Properties & Events :

AboutComponent	Component Information
Attributes	Attributes of the selected File
AttributesShort	Short Version of the Attributes (OLD DOS STYLE)

Date	File Date
Time	File Time
Size	File Size
Path	FilePath
NameOnly	Name of the File without the Path
NameandExtension	Name and Extension of the File
Extension	Extension of the File
Filename	Set this property to change the selected File

TZipFileView

TZipFileView = Class(TComponent)

A Component that collects Information about a ZipFile and its contents. To use it drop it onto your Form. Enable the Information you want to collect about the contents . It will display all Items found in the Items property.   The Delimiter will be used to show the different properties per Filename.

Properties & Events :

AboutComponent	Component Information
Delimiter	A Delimiter for the different Information
Filename	The Zip File about which you collect the Information
Items	A List of Information returned
ViewOptions	A List of Options on the type of Information to collect

TDriveVolumeInfo

TDriveVolumeInfo = Class(TComponent)

TDriveVolumeInfo supplies you with Information about your Drive Volume like the Volume Label, the serial Number, the type of File System etc. . This Component is an Idea of one of the Users of the AHM Component Collection. Thanks goes to him and his need of getting the Serial Number of a Drive. Lets hope its helping other people now who are in need of the same Information.

Properties & Events :

AboutComponent	Component Information
CasePreserved	Case is preserved
CaseSensitive	Case is sensitive
CurrentDrive	This is the property that decides for which drive the information is displayed. Set it to any available Drive and it will change the Information
FileCompression	Compression is enabled
FileSystemFlags	Flags of the FilesSystem
FileSystemName	Type of File System
PersistentACLS	ACLS is persistent
SerialNumber	Drives Serial Number
StoredUniCodeOnDisk	Uni - Code is stored on the Drive
VolumeIsCompressed	Indicates if the Volume is a compressed Drive
VolumeLabel	the VolumeLabel of the Drive

TMultiFileLabel

TMultiFileLabel = Class(TCustomLabel)

This is a Label Component that will display a Filename on it. If the Filename is too long for the Label it will take out the Path and specify it eg. D:\...\myfile.txt. If the Name is still too long for the label it will drop the

Driveletter eg. ...\myfile.txt and if the name is still too long for the Label it will show as myfile.txt. All that is done using the normal **TMultiLabel** style.

Properties & Events :

AboutComponent	Component Information
Filename	Name of the File which is about to be displayed
FontBackground	Background on the Font
Background	Background of the Label
Font3D	Three-dimensional FontStyles
Frame	Frame around the Control

TFormExplode

TFormExplode = Class(TComponent)

If dropped onto your Form this Component will show the form in a kind of exploding style.   There are 19 different Styles to show this event and you can choose different ones for the onclose and onshow event.
If you still not satisfied you could choose any point on the screen from which this Form is gone explode from. If that is still not enough then you can choose the OnGetPoints event and assign your own points there to the origin.

Properties & Events :

AboutComponent	Component Information
CustomLeft	Left point to use as Origin if style is wsCustomPoint
CustomRight	Right point to use as Origin if style is wsCustomPoint
CustomTop	Top point to use as Origin if style is wsCustomPoint
CustomBottom	Bottom point to use as Origin if style is wsCustomPoint
NoOfSteps	Number of Steps which it uses to explode
OnCloseType	Specify the Type of Origin when the Form closes
OnShowType	Specify the Type of Origin when the Form shows
ShowType	Specify when the effect is going to apply itself eg. Never, On Close only, On Show only
OnGetPoints	specify even different Points for the Origin

TFormTopMost

TFormTopMost = Class(TComponent)

If this Component is set to Active your Form will float on top of all other Forms. Nice for implementing those floating toolwindow effects.

Properties & Events :

AboutComponent	Component Information
Active	Enables/.Disables the Floating

TFormSizeOptions

TFormSizeOptions = Class(TComponent)

This Component will limit your Size to the amounts indicated.

Properties & Events :

AboutComponent	Component Information
MinimumHeight	Sets the MinimumHeight
MinimumWidth	Sets the MinimumWidth
MaximumHeight	Sets the MaximumHeight
MaximumWidth	Sets the MaximumWidth
Options	Enable/Disable the different Limitations

TLogFile

TLogFile = Class(TComponent)

A Component that will log everything in gets passed into a Textfile. Good for debugging your Application and creating Logfiles of Users logged onto your Application etc. .

Procedures & Functions :

Procedure Log(LogStr : String);	the logging Call
Procedure ClearLog;	Clears the Logfile

Properties & Events :

AboutComponent	Component Information
Header	The Header in the File
LogDate	Logs the Date with each Line when Enabled
LogFile	Name of Log File to create
LogHeader	Creates the Header if enabled
LogSpecial	Logs a special String with each Line e.g. Username
LogTime	Logs the time with each log even when enabeld
Special	the Special String to Log

TCheckbox3D

TCheckbox3D = Class(TCustomControl)

Checkbox with the enablement for 3D Captions, Buttontypes (Normal, Borland Pascal Type, As Button) and a Background using the Background Property. Different Color for you checkbox and your checkmark. This is truly a new definition of a Checkbox.

Properties & Events :

AboutComponent	Component Information
Caption	Caption on the Checkbox
CheckMarkColor	Color of the Checkmark
CheckBoxColor	Color of the box where the Checkmark is placed
Background	Background on the whole Control
Checked	Indicates if the Checkbox is checked by default
Down	Indicates if the Checkbox is checked in the group
Font3D	Threedimensional FontStyles
TransparentCheckbox	Indicates the Transparency of the Background
State	Indicates the state the Checkbox is currently in
ButtonType	Choose a predefined Button Type between the three different Styles : • Normal • Borland Pascal 7.0 • As Button

TRadioButton3D

TRadioButton3D = Class(TCustomControl)

Radiobutton with the enablement for 3D Captions, Buttontypes (Normal, Borland Pascal Type, As Button) and a Background using the Background Property. Different colors for your Radiobutton and your Checkmark. This is truly a new definition of a Radiobutton.

Properties & Events :

AboutComponent	Component Information
Caption	Caption on the Radiobutton
CheckMarkColor	Color of the Checkmark
CheckBoxColor	Color of the box where the Checkmark is placed
Background	Background on the whole Control
Checked	Indicates if the Checkbox is checked by default
Down	Indicates if the Checkbox is checked in the group
Font3D	Threedimensional FontStyles
TransparentCheckbox	Indicates the Transparency of the Background
State	Indicates the state the Checkbox is currently in
ButtonType	Choose a predefined Button Type between the three different Styles : • Normal • Borland Pascal 7.0 • As Button

TMultiPresentation

TMultiPresentation = Class(TCustomLabel)

After all this time this is proberly one of the better Components that have come out now developed by me. (Hehehe ... personal believe) This Component was written inspired by whatever FX released as Freeware on the Internet. Or lets say I had a look at one Procedure of it. Very interesting were the first Tests in Delphi 3 where I suddenly only saw parts of the Picture and realized that the Procedure itself needed to be completely rewritten. Well I have done it and hereby present it to the domain as Part of my Collection. This Component does the almost easy implementation of adding slides to your Software. Just set your Backgrounds (for the Background and the Font), set the Font3D Style,
and all other Settings as you prefer them .
I have added and exchanged many Styles to the original so please don’t be suprized it this Component has nothing at all to do with FX’s release anymore.

After you have configured the look for your Slides you can start adding them and here are the Steps :

- Add Slide
- Indicate Empty,Text,Bitmap
- Insert your Text or Your Bitmap
- Choose the Effect (out of 40 to be available)
- Done

Procedures & Functions :

Procedure LoopSlides;	Starts the Presentation
Procedure Stop;	Stops the Presentation

Properties & Events :

AboutComponent	Component Information
Background	Background of the Label
Font3D	Threedimensional FontStyles
FontBackground	Background on the Font
Slides	The Slides Property where you can add Slides

TMultiLabel

TMultiLabel = Class(TCustomLabel)

Another Version of the Label Component. This time the Background is drawn on the actual Font.
The Heading in this Helpfile (AHM Component Collection) is designed using TMultilabel.
The Problem with the Color is finally fixed ... you can now set it to any Color.

Properties & Events :

AboutComponent	Component Information
AsButton	Label becomes a button if set to true
FontBackground	Background on the Font
Background	Background of the Label
Font3D	Three-dimensional FontStyles
Frame	Frame around the Control

TMultiBtn

TMultiBtn = Class(TCustomControl)

The Multibutton is a new type of button which lets you specify almost everything on the Look and Feel of the Button. You can set the Background and the Font to almost any look you can imagine.

Properties & Events :

AboutComponent	Component Information
Font3D	Threedimensional FontStyles
Bitmap	Insert Bitmap to be used as Glyph, the Buttone will automatically gray the Bitmap when it is disabled
Background	Background of the Label
WordWrap	enabled wordwrapping of the caption on the Button

TMultiHeader

TMultiHeader = Class(TCustomControl)

A Header Control with Background and 3D Fonts. For potential use have a look at THeaderList.

Properties & Events :

AboutComponent	Component Information
Font3D	Threedimensional FontStyles
Background	Background of the Label

TOffice97Edit

TOffice97Edit = Class(TEdit)

The MS-Office 97 Version of an Edit Box. When the Edit Box gets the Focus its Frame changes to indicate that it has the focus. Also available as Data - Aware Version in **TDBOffice97Edit**.

Properties & Events :

AboutComponent	Component Information

TDBOffice97Edit

TDBOffice97Edit = Class(TDBEdit)

The MS-Office 97 Version of an Edit Box. When the Edit Box gets the Focus its Frame changes to indicate that it has the focus. Also available as Non - Data - Aware Version in **TOffice97Edit**.

Properties & Events :

AboutComponent	Component Information

TClipboard

TClipboard = Class(TScrollBox)

A Component that automatically displays the Contents of the Clipboard. It has a slight problem **ONLY** in designtime once you have run your programs of displaying the contents but when you refresh that form it will display right again.

Properties & Events :

AboutComponent	Component Information

THeaderList

THeaderList = Class(TCustomControl)

A Header Control with Background and 3D Fonts connected to a Listbox that supports several Rows next to each other delimted by any Character in the actual List.

Properties & Events :

AboutComponent	Component Information
HeaderFont3D	Threedimensional FontStyles
Background	Background of the Label
Delimiter	Delimiter for delimiting the rows in the List
HeaderSections	Row Headers

TIE3DButton

TIE3DButton = Class(TGraphicControl)

Have you seen Internet Explorer ? Well those Buttons on top is what we have here. I have also added 3D Captions to it since I try to support that everywhere now. Have fun playing around with it. This Version of it has undergone major changes and still I have received feedback in the Beta Version that there were problems with
some of the displaying option. Will I ever get this one right ?
I hope that it will now do exactly what it is supposed to do. Here are the options not inherited from TGraphiccontrol.

Properties & Events :

AboutComponent	Component Information
Alignment	Aligns the Button
AllowAllUp	Indicates if all Buttons are allowed to be up in a Button Group
BevelLooks	Specifies the Border of the Button - lk97 - lkDeep - lkNormal
Bitmap	Bitmap of the Button when it has Focus
DesignTimeFrame	Enables/Disables a Frame around the Button in Designtime for means of visibly aligning a Group of Controls
Font3D	3D Font of this Button
GroupIndex	Index in a group of Buttons
DisabledBitmap	Bitmap when the Button is Disabled
Down	Indicates if the Button is down by default
DropDownMenu	PopupMenu that will dropdown when the Button is pressed
Enabled	Disabled/Enabled
Layout	presents you with a list of choices as to where to place your Bitmap -blBitmapTop .. places it on Top -blBitmapLeft .. places it on the Left -blBitmapRight .. places it on the Right -blBitmapBottom .. places it on the Bottom

+Options

opMonodisplay Displays a monochrome Version of your Bitmap while the Button has no Focus

opMenuArrow if a dropdownmenu is attached it will display a Menuarrow on the right side of your Control

opTextonly If set to true will display Text only

opBitmaponly If set to true will display the Bitmap only

ShadingType indicates how to shade the monochrome Version of your Bitmap when the Button is not focused

TMultiPanel

TMultiPanel = Class(TCustomPanel)

TMultiPanel gives you the ability to nearly totally modify the Panel. The Panel can now have nearly any look and feel. All other properties are inherited from TCustomPanel. I have changed the orginal Bevelproperties for this component to give you the ability to add your own colors and styles to the Bevels.

Properties & Events :

AboutComponent	Component Information
Font3D	Threedimensional FontStyles
Background	Background of the Label
DlgFrame	Adds a raised Dialog Form Look to the Panel
FrameInner	Inner Bevel Look and Feel
FrameOuter	Outer Bevel Look and Feel

TMultiGauge

TMultiGauge = Class(TGraphicControl)

A gauge component which you can now also give almost any look an feel. This one has replaced the original TFlexigauge and improves not only on the functionality but also on the graphical interface. If you want a see-through Border just dont specify a Background.

Properties & Events :

AboutComponent	Component Information
Picture	A Bitmap that can be attached to one of the sides of the gauge
Font3D	Three-dimensional FontStyles
Background	Background of the Label
BevelInner	Draws the inner Frame
BevelOuter	Draws the outer Frame
PicturePosition	Position of the picture attached to the Gauge
Borderwidth	Width of the Border between Inside and Outside Bevel
Kind	Horizontal or Vertical look and feel
SegmentGap	width of the Segment Gap
SegmentWidth	width of the Segment
SegmentColor	Color of the Segment
InsideColor	Color of the Gauge Background
Percantage	Indicates the Percentage

THints

THints = Class(TComponent)

A New Hint Box Display. Once you drop it onto your form all Application Hints will be displayed in the Format you specify. A 3D Hint that can be displayed as a Speechbubble if the need may be.
This Component also includes an Editor for MultiLine Hints which will be automatically attached to all Controls which have Hints. New Properties have been added Play around and you should get to know them.

Properties & Events :

AboutComponent	Component Information
Font3D	Threedimensional FontStyles
Speechbubble	Makes the Hint look like a Speechbubble
Shadow	Specifies if the Hint has a Shadow
HintDirection	Specifies the Direction of the Hints Speechbubble
UnusedColor	Specifies a unused Color in the Hint.Needed for drawing Operations !
HintShadowColor	Color of the Shadow
HintFont	Font used for the Hint
HintPauseTime	Time the Hint Delays on the Screen
OnSelectHintDirection	Event that executes when the Hint Direction changes
Background	Background of the Label

TColorBevel

TColorBevel = Class(TBevel)

A Bevel with the ability to change the used Colors.

Properties & Events :

AboutComponent	Component Information
Color1	Highlighted Color
Color2	Shadow Color

TColorCombo

TColorCombo = Class(TCustomComboBox)

This Component lets you select Colors out of a Combobox. I saw this Component out there and tried to figure it out myself, well here it is. Property Colorselected is used for reading the currently selected color.

Properties & Events :

AboutComponent	Component Information
SelectedColorname (Runtime ReadOnly)	Returns the name of the Selected Color
ColorSelected	Returns and sets the name of the Selected Color
ColorDisplaywidth	Width of the Color Bar
OnGetColorNum	
OnGetColor	
ColorFrame	Bevel around the Color Bar
Font3D	Threedimensional FontStyles

TFontCombo

TFontCombo = Class(TCustomComboBox)

This Component lets you select Fonts out of a Combobox. I saw this Component out there and tried to figure it out myself, well here it is. Property Fontselected is used for reading the currently selected font.

Properties & Events :

AboutComponent	Component Information
FontSelected	Returns and sets the currently Selected Font

TFillBackground

TFillBackground = Class(TGraphicControl)

A new improved Gradientfill Component which you now can use for almost everything. I have attached even the ability to create tiled Bitmaps and other features which come by default with the Background Component..

Properties & Events :

AboutComponent	Component Information
Background	Background of the Label

TWinStringGrid

TWinStringGrid = Class(TStringGrid)

A Floating Stringgrid. Use this Component if you want to enable the Grid to float on top of your Window.

Properties & Events :

AboutComponent	Component Information

TSystemMenu

TSystemMenu = Class(TComponent)

This is another add on to your Menu Collection. If you want to add Items to your Forms Menu or to your Applications Menu (Latter is only available when you mimize your Application).

Properties & Events :

AboutComponent	Component Information
AttachTo	which Menu you want to attach the Item to
Bitmap	Bitmap to show your Item
Checked	Checked or Unchecked RadioItem
CheckedGlyph	The Glyph to indicate the checked state
Enabled	Enabled or Disabled Item
MenuItemType	Type of Item to add
MenuName	Name for the Item on the Menu
PositionOnMenu	where to position the Item
RadioItem	Indicates the RadioItem state
UnCheckedGlyph	The Glyph to indicate the unchecked state
UniqueMenuID	Id on the Menu

TWinCheckboxList

TWinCheckBoxList = Class(TCustomListBox)

A floating Listbox consisting out of Checkboxes which can be checked for the currently selected Item.

Properties & Events :

AboutComponent	Component Information
Checked [Indexed]	Indicates if the Item is checked *Runtime read-only
State [Indexed]	Indicates the State the Item is in *Runtime read-only
AllowGrayed	
Offset	Offset from the Checkbox
ColorCheckedFont	Color of the Font when the Box is checked
Font3D	Three-dimensional FontStyles
Float	Creates a floatable/moveable Window around the Listbox

TAsciiDBtransfer

TAsciiDBTransfer = Class(TComponent)

Component to export/import your Data. One of many out there. This one is designed and developed by me for the use of one of my own Applications. After it successfully worked I decided to include it into the collection.

Procedures & Functions :

Procedure Export;	Starts the Export
Procedure Import;	Starts the Import

Properties & Events :

AboutComponent	Component Information
DataSource	DataSource to connect to
Delimiter	Delimiter to use in the Export/Import
Fields	Settings for the Fields to be exported/imported
Filename	Filename to export to /import from
PrintHeader	Disable the printing of a header in the file

TWinListBox

TWinListBox = Class(TListBox)

Properties & Events :

AboutComponent	Component Information

TColorDice

TColorDice = Class(TGraphicControl)

A Dice Component for all your Games. Use the Background property to create marble, wooden or even golden Dices for the Games you want to create.

Procedures & Functions :

Procedure Roll;	will roll the dice and return with a different random face

Properties & Events :

NormalBackground	Background when Dice is not on Hold
HoldBackground	Background when Dice is on Hold
Value	Value of the Dice (1..6)
Hold	Indicates whether or not the Dice is being held

DotColor	Color of the Dots on the Dice
AboutComponent	Component Information

TLed

TColorDice = Class(TGraphicControl)

Another Led Component. In future this one will get the Background Property.

Procedures & Functions :

Procedure LedOn;	Turns Led On
Procedure LedOff;	Turns Led Off
Procedure Blink;	Enables/Disables the blinking

Properties & Events :

LedisOn	Indicates if the Led is on
LedOnColor	Color of the Led when Led is on
LedOffColor	Color of the Led when Led is off
LedShadow	Shadow of the Led
AboutComponent	Component Information
Interval	Interval the Led has to change Color when Blinking
Blinking	Indicates if the Led is currently Blinking

TDigital

TDigital = Class(TGraphicControl)

A Digital Number Display Component. Well play around with it. It's good to use in Games. Got the Idea from one of my DOS Games.

Properties & Events :

AboutComponent	Component Information
MaxDigits	Maximum amount of Digits
Number	the number displayed
DigitOnColor	Color of the Digits
DigitOffColor	Color of the off Digits
DigitBackground	Color of the Control
Border3D	the 3D Border around the Control
OnNewNumber	event on a new Number

TJokeHolder

TJokeHolder = Class(TComponent)

A Jokeholder Component that is very useful when you want to keep lines of information. Use this one as the TipTrickholder or similar.

Procedures & Functions :

Function Getany : String;	Gets any String from the List of Items
Function Getone (Which : Integer) : String;	Gets a specific String from the List of Items

Properties & Events :

AboutComponent	Component Information
Items	List of Jokes

TOwnerShip

TOwnerShip = Class(TComponent)

A simple Component storing Ownership Information. I plan to remove this Component sometime in Future and will release it as Freeware then. This Component is very simple and should not encourage me to explain it too well.

Properties & Events :

AboutComponent	Component Information
Name	Name
POBox	POBox
TelephoneNo	Telephone Number
FaxNo	Fax Number
Street	Street
Town	Town
Postal Code	Postal Code
Special1	Extra String
Special2	Extra String
Special3	Extra String

TStringHolder

TStringholder = Class(TComponent)

A simple Component to store a list of Items.

Properties & Events :

Items

AboutComponent	Component Information

TSpeakerSound

TSpeakerSound = Class(TComponent)

The old functionality of Sound through the Speaker from Turbo Pascal integrated into Delphi. Watch out in Windows NT. This Component directly accesses the Hardware which Windows NT doesn’t like too much. Please only use it when developing for Windows 95. I am still trying to figure out a Workaround for the actual Problem, yet it will be a while before I get around to fix it up completely.

Procedures & Functions :

Procedure Delay(MSecs: Integer);	Delays the Sound for the specified Number of Milliseconds
Procedure Sound(Freq: Word);	Starts the Sound with a pitch indicated in Freq.
Procedure NoSound;	Switches the Sound off

Properties & Events :

AboutComponent	Component Information
Soundenabled	Can be used to disable all calls to sound
Delayenabled	Can be used to disable all calls to delay

TAnimator

TAnimator = Class(TWinControl)

This Component enables playing of AVI Files.

Properties & Events :

AVIFileName	The avi to be played
AboutComponent	Component Information
Center	Centers the display of the avi in the middle of the Component
Transparent	Indicates the transparency
PlayFrom	Play only from a certain Part of the Movie
PlayTo	Play only to a certain Part in the Movie
Cycles	Indicates how many times the Movie cycles through (-1 means constantly)
Action	Action on the Avi File
AutoAnimate	Indicates if the movie automatically animates itself

TImage3D

TImage3D = Class(TGraphicControl)

An Image Component which draws a 3D Border around the Image using the actual Image as the Background.

Properties & Events :

AboutComponent	Component Information
Picture	Picture to be used
Intensity	Intensity of the Border higher Numbers mean come nearer to black or white
Center	Centers the Image in the Control
Border3Din	Indicates if the Bevel is raised or lowered
Borderwidth	Width of the 3D Border

TImagescroller

TImageScroller = Class(TGraphicControl)

This component will scroll Text or even Images across the Background of your form. It is a nice little component to create Creditscreens or display About Information that will catch the Eye.

Properties & Events :

AboutComponent	Component Information
Scrolling	Indicates if the Control is scrolling
Text	Text to be scrolled
Bitmap	Image to be scrolled
ScrollDirection	Direction to be scrolled
ScrollInterval	Delay between the scrolls
ScrollStep	Amount of Pixels to skip
ScrollInverse	Enable Inverse Scrolling
Font3D	Three-dimensional FontStyles
Pause	Pauses the Scrolling
TextWindowLeft	Left Position of the Text
TextWindowTop	Top Position of the Text
TextToCenter	Centers the Text

TFadingLabel

TFadingLabel = Class(TCustomLabel)

A Label Component that fades its contents in and out respectively.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
Fade	Starts fading when Set to true
BetweenColors	Number of Colors while fading
FadeAllTheTime	enables/disables continuously running
FadeSpeed	Speed at which to fade the Text
OnNotVisible	Event when Text is not visible
OnTotallyVisible	Event when Text is completely visible

TDBSearchForm

TDBSearchForm = Class(TComponent)

A Lookup Component which creates a form that lets you search for certain values in the dataset. Typically would you use it in conjunction with the TNavigationForm in the Onfind event to locate certain Data in the Current Table. A index is required to be defined on the searched table. A simple lookupcombobox is very good when it comes to small amounts of data, yet when you need huge amounts of data to look up upon then it is the best Idea to implement a seperate form to search on.

Procedures & Functions :

Procedure Execute;	Calls the Searchform to display
Function Return (Fieldname,Nowvalue : String) : String;	Calls the Searchform using the Fieldname for the Index and Nowvalue for the currently selected Item in the list

Properties & Events :

AboutComponent	Component Information
Datasource	Indicates the Datasource it is attached to read from
FormCaption	Caption of the Form
FormPosition	Position of the Form on the Screen
FormFonts	Fonts to be used on the Form
BtnCancelGlyph	Glyph for the Cancel Button
BtnSelectGlyph	Glyph for the Select Button
BtnSearchGlyph	Glyph for the Search Button
DsgnExecute	Execute in Designtime
BtnCancelName	Caption of the Cancel Button
BtnSelectName	Caption of the Select Button
BtnSearchName	Caption of the Search Button
SelectionComboVisible	Visibility of the Combobox
Searcheditvisible	Visibility of the Editbox
ButtonPanelVisible	Indicates if the Panel with the Buttons Select and Cancel is visible
SelectionLabel	Caption of the Selection Label
SearcheditLabel	Caption of the Editbox Label
OnCancel	
OnSelect	
FormTop	Position of the Form y-axis on the Screen
FormLeft	Position of the Form x-axis on the Screen
FormWidth	Width of the Form
FormHeight	Height of the Form

TDBPasswordDlg

TPasswordDlg was developed for all those people that have Usertables in their Database. Attach it to a Datasource and set the Fields for the Username and the Password. Then call execute from your mainform after you open your Database (e.g. Database1.open). Now set the events of what you want to do.

Properties & Events :

AboutComponent	Component Information
DataSource	Datasource pointing to the UserTable
DataPassword	Field of the stored Passwords
DataUsername	Field of the stored Username
Fonts	Font used on Password Dialog
PasswordChar	Character used for the Password Display (usually Asterix)
GlyphCancel	Glyph used on the Cancel Button
GlyphOk	Glyph used on the Ok Button
DialogCaption	Caption used on the Dialog
EditCaption	Caption of the Editbox
Sound	Sound to play when the Dialog comes up
OnRightPassword	Event executed when the user has entered the right Password
OnWrongPassword	Event executed when the User entered the wrong Password
OnOk	Event executed when the User presses Ok
OnCancel	Event executed when the User presses Cancel
Scramble	Property that hold the same Information as the Password Property scrambled
Password	Property that holds the Password the Dialog will be looking for when executed
DsgnExecute	Property to execute Dialog in designtime
NoOfRetries	No of times the User can retype the Password wrongly before the Afterretries event is called
Afterretries	Event that executes after the user has unsuccessfully entered the Password a number of times per Dialog call.

TDBMultiNavigator

TDBMultiNavigator = Class(TGraphicControl)

A DBNavigator with some extras. You can set it to display itself vertical and if you enable the Flat property then you can set a Background Style.

Properties & Events :

AboutComponent	Component Information
Background	Background when the Buttons are Flat
Direction	Vertical or Horizontal Navigator

TDBIndexLookup

TDBIndexLookup = Class(TCustomCombobox)

A Database Combobox that displays and sets the current Index on your Table.

Properties & Events :

AboutComponent	Component Information
DataSource	Datasource pointing to the Table
DisplayType	Type of Display used for the Lookup
NoIndexEnabled	Used when you want to enable a non-indexed Table
NoIndexItemName	Name of the Item to be added

TWinDBListbox

TWinDBListBox = Class(TDBListBox)

A Floating Database Listbox.

Properties & Events :

AboutComponent	Component Information

TPasswordDlg

TPassWordDlg = Class(TComponent)

A Password Dialog that verifies who logs into your Application. This Component also has a Property Scramble which Scrambles the Password for reading and writing to Inifiles.

Procedures & Functions :
Function Execute : Boolean;

Properties & Events :

Fonts	Font used on Password Dialog
PasswordChar	Character used for the Password Display (usually Asterix)
GlyphCancel	Glyph used on the Cancel Button
GlyphOk	Glyph used on the Ok Button
DialogCaption	Caption used on the Dialog
EditCaption	Caption of the Editbox
Sound	Sound to play when the Dialog comes up
OnRightPassword	Event executed when the user has entered the right Password
AboutComponent	Component Information
OnWrongPassword	Event executed when the User entered the wrong Password
OnOk	Event executed when the User presses Ok
OnCancel	Event executed when the User presses Cancel
Scramble	Property that hold the same Information as the Password Property scrambled
Password	Property that holds the Password the Dialog will be looking for when executed
DsgnExecute	Property to execute Dialog in designtime
NoOfRetries	No of times the User can retype the Password wrongly before the Afterretries event is called
Afterretries	Event that executes after the user has unsuccessfully entered the Password a number of times per Dialog call.

TTipTrickDlg

TTipTrickDlg = Class(TComponent)

A TipTrickDialog like the default one that comes with Windows. This one you can configure for use with your own Tips for your Application.

Procedures & Functions :
Procedure Execute;

Properties & Events :

SetupCheckBox	Properties for the Checkbox
Custom1BitBtn	Properties for the first Custom button
Custom2BitBtn	Properties for the second Custom button
OkBitBtn	Properties for the Ok Button
NextTipBitBtn	Properties for the Next Button
Tips	Tips for the TipTrick Dialog
OnCustom1	Event that executes when the first Custom button is pressed
OnCustom2	Event that executes when the second Custom button is pressed
OnOk	Event that executes when the Ok Button is pressed
OnNext	Event that executes when the Next Button is

	pressed
OnCheckClick	Event that executes when the Checkbox is clicked
AboutComponent	Component Information
DsgnExecute	Event to execute in designtime
TipDlgCaption	Caption of the TipDialog
DidYouKnow	Caption of the DidYouKnow Label
Fonts	Fonts used in the Dialog

TTimePopup

TTimePopup = Class(TComponent)

A Time Selection Component that shows a Dialog that will return a string containing the time selected.

Procedures & Functions :

Function Execute : String;

Properties & Events :

HourFrom	the hour the dialog ranges from
HourTo	the hour the dialog ranges to
MinutesFrom	the minutes the Dialog ranges from
MinutesTo	the minutes the Dialog ranges to
MinutesIncrease	the differences between the selectable Minutes e.g. 5 for 5 minutes
DsgnExecute	Execute the Component in Designtime
ReturnOnCancel	String to return when the Dialog is canceled without selecting a time
AboutComponent	Component Information

TDatePopUp

TDatePopup = Class(TComponent)

A Date Selection Component.

Procedures & Functions :

Function Execute : String;

Properties & Events :

AboutComponent	Component Information
Month	the month returned
Day	the day returned
Year	the year returned
ReturnOnCancel	date that should be returned when the dialog is canceled
Return	the return of Function execute on cancel
Fonts	Font used in the Dialog
DsgnExecute	Execute the Component in Designtime
UseDefaultDate	Specifies if the default selected Date is the current date or not
OnDateSelect	Event that executes when a date is selected and returned
DefaultGlyph	Specifies if the Dialog uses the default Glyphs of the Component
GlyphPrevYear	Glyph on Previous Year Button
GlyphPrevMonth	Glyph on Previous Month Button
GlyphOk	Glyph on Ok Button
GlyphCancel	Glyph on Cancel Button
GlyphNextMonth	Glyph on Next Month Button
GlyphNextYear	Glyph on Next Year Button

TMsgBox

TMsgBox = Class(TComponent)

A Wrapper around the MsgBox Functions which also enables MultiLine Text, Sound , executing in Designtime and many more.

Procedures & Functions :

Function Execute : Word;

Properties & Events :

AboutComponent	Component Information
MessageText	Lines of text to show when the Dialog is executed
MessageType	Type of Message
DsgnExecute	Execute in Designtime
Buttons	Buttons on the Dialog
MsgReturn	Specifies the button last pressed when executed
HelpContext	Help Context for the Helpbutton
Xposition	Position of the Dialogs x-axis on the Screen
Yposition	Position of the Dialogs y-axis on the Screen
Sound	Sound the Dialog makes when it executes
OnYes	Event on the relevant button pressed
OnNo	Event on the relevant button pressed
OnOk	Event on the relevant button pressed
OnCancel	Event on the relevant button pressed
OnAbort	Event on the relevant button pressed
OnIgnore	Event on the relevant button pressed
OnRetry	Event on the relevant button pressed

OnAll	Event on the relevant button pressed

TFolderBrowser

TFolderBrowser = Class(TComponent)

A Wrapper around the Windows Function to browse for Folders on your current System.

Procedures & Functions :
Function Execute : Word;

Properties & Events :

AboutComponent	Component Information
BrowseCaption	Caption for the Dialog
BrowsePathLabel	Displays a Label with the Path in the Dialog
DsgnExecute	Execute in Designtime
BrowseReturnPath	Path returned on execute
BrowseRoot	a list of possible Options where the Dialog begins to display its Folders
BrowseFlags	Options on how to display the Dialog

TScreenCanvas

TScreenCanvas = Class(TComponent)

A Wrapper around the Windows Canvas. Now you can draw on it directly by just accessing it as you would with a normal Canvas.

Procedures & Functions :
procedure RestoreScreen;

Properties & Events :

AboutComponent	Component Information
Canvas	the Canvas property of type TCanvas
AutoRestores	Restores the Screen automatically once your Program ends.

TMsgDlg

TMsgDlg = Class(TComponent)

A Wrapper around the MsgDlg Functions in Delphi. Also enables Sound and instant feedback.

Procedures & Functions :
Function Execute : Boolean;

Properties & Events :

InputHeader	Caption of the MsgDialog
InputQuestion	Caption of the Question
InputReturn	Return on Execution
AboutComponent	Component Information
DsgnExecute	Execute in Designtime
Sound	Sound the Dialog makes when it executes
OnOk	Event that executes on Ok
OnCancel	Event that executes on Cancel

TAboutInformation

TAboutInformation = Class(TComponent)

Another AboutInformation Component

Procedures & Functions :
Procedure Execute;

Properties & Events :

VersionDescription	Version
ActualInformation	Lines of text of actual Information
SystemInformation	Lines of text of systeminformation e.g. You can specify the Information about the current User logged in etc.
DsgnExecute	Execute in Designtime
Fonts	Fonts in the Dialog
AboutComponent	Component Information

TAccess

TAccess = Class(TComponent)

An Access Component to enable/disable access to the application. Similar you could use TPasswordDlg or TSerialNumber.

Procedures & Functions :
Function Execute : Boolean;

Properties & Events :

AboutComponent	Component Information
AccessCode	Access Code for the Application
OnWrongCode	Event on wrong Code
OnRightCode	Event on right Code

TQuickInfo

TQuickInfo = Class(TComponent)

Component to supply quick information about something specific to the User of your Application

Procedures & Functions :
Procedure Execute;

Properties & Events :

Alignment	Alignments of the Lines
Line1	Line of text
Line2	Line of text
Line3	Line of text
Line4	Line of text
Line5	Line of text
Line6	Line of text
Line7	Line of text
Line8	Line of text
Line9	Line of text
Line10	Line of text
Infoboxcaption	Caption of the Infobox
AboutComponent	Component Information
Fonts	Fonts used in Dialog
DsgnExecute	Execute in Designtime

TLogWindow

TLogWindow = Class(TComponent)

Component that logs information to a Window.

Procedures & Functions :

Procedure Execute	Calls the Window
Procedure Log(Value : String);	Logs a value
Procedure Clear;	Clears the Window

Properties & Events :

AboutComponent	Component Information
LogWindowCaption	Caption of the Window
LogWindowColor	Color of the Window
LogWindowFont	Font of the Window
DsgnExecute	Execute in Designtime
FileMenu	Has a Filemenu
OpenFilename	Name of File to open by default
OpenFileOnExecute	Open the file by default
OnExecute	Event on execute
OnExit	Event on exit
OnSaveFile	Event on Save successfull executed
WindowState	State of the Logwindow
WindowPosition	Position of the Window
BorderIcons	Icons used in the Border
WinddowWidth	Width of the Window
WindowHeight	Height of the Window
ReadOnly	Text in Window is read-only
LoadWindowIcon	Icon for Window

TShellAbout

TShellAbout = Class(TComponent)

A Wrapper around Windows own About Dialog.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
DsgnExecute	Execute in Designtime
TitleLine	Line of the Title of the dialog
InfoLine	Line of Application Information
FirstLine	Line of Details

TSplashScreen

TSplashScreen = Class(TComponent)

Have a look what you doing while using the new execute procedure in Designtime. Once you finished call execute from your Mainform.OnCreate event and call close in your project.pas file just before the Statement Application.run in the following way :

- Mainform.Splashscreen.close;

Once you have done that you have your Splashscreen !

Procedures & Functions :

Procedure Execute;	Execute the Splashscreen
Procedure Close;	Close the Splashscreen

Properties & Events :

Image	Image to use
OnRun	Event executed on running of the Splashscreen
SplashWidth	Width of the Splashscreen
SplashHeight	Height of the Splashscreen
OnSplashDone	Event executed when the splash is done
DelayInMsec	Delay that the Splashscreen stays visible at least
AboutComponent	Component Information
DsgnExecute	Execute in Designtime
Background	Background of the Splashscreen
OutSideFrame	Outsideframe of the Splashscreen
InsideFrame	Insideframe of the Splashscreen
Borderwidth	Width of the Border
ScreenCenter	SplashScreen is Centered in the Mainscreen

TSerialNumber

TSerialNumber=Class(TComponent)

A SerialNumber Component an Idea of Michael Wakkinen, one of my registered Users.
Now you can protect your Programs against non-registration. Just use this Component as is. Execute it from somewhere and it should ask you to specify your SerialNumber.
Use CalcOwnerName and CalcSerialNumber in Designtime only to calculate you the SerialNumber from the Ownername.Use Suffix and Prefix as variants in the SerialNumber. Use Uniquefivedigits if you want to obtain a different SerialNumber for all your programs/Versions. Now put it in your Software and once someone wants to register just tell him to send you his name and then you obtain the Serialcode by using the same UniquefiveDigits,Prefix,Suffix and you can send him the Code to run your Application fully.

Procedures & Functions :

Function Execute : Boolean;

Properties & Events :

AboutComponent	Component Information
Prefix	Unique Prefix used in the Serial Number
Suffix	Unique Suffix used in the Serial Number
UniqueFiveDigits	Unique 5 Digits used for the actual calculation
CalcOwnerName	Use this in Designtime only
CalcSerialCode	Use this in Designtime only
OnWrongCode	Event that executes on the Wrong Code
OnRightCode	Event that executes on the Right Code
CloseInputOnWrongCode	Closes the Input on Wrong Code after executing event OnWrongCode
OnClose	Event on close
ButtonOk	Properties for the OK Button
ButtonCancel	Properties for the Cancel Button
Font	Font used in the Dialog
NameCaption	Caption of the Name Editbox
SerialCaption	Caption of the Serial Number Editbox
DsgnExecute	Execute in Designtime

TIniEntry

TIniEntry = Class(TComponent)

A Wrapper around the Inifile. Use this Component to store Information about your Application in an Inifile. The Name variable refers to the name of the entry and Value to the Value of the Identifier.

e.g. Name=Value

Procedures & Functions :

Function RStr(Name : String):String;	Read a String from the Inifile
Procedure WStr(Name,Value : String);	Write a String to the Inifile
Function RCryptStr(Name : String):String;	Read and decrypt a String from the Inifile (Passwords for example)
Procedure WCryptStr(Name,Value : String);	Write and encrypt a String to the Inifile
Function RInt(Name : String):Integer;	Read a Integer from the Inifile
Procedure WInt(Name : String;Value : Integer);	Write a Integer to the Inifile
Function Rbool(Name : String;):Boolean;	Read a Boolean from the Inifile
Procedure WBool(Name : String;Value : Boolean);	Write a Boolean to the Inifile
Procedure Erase;	Erase the whole Section in the Inifile

Properties & Events :

FileName	Name of the Inifile
Section	Section in the Inifile (if it doesn’t exist it will be created)
OnError	Event that occurs if one of the functions returns an error
AboutComponent	Component Information

TTimeTrigger

TTimeTrigger = Class(TComponent)

Use this Component to trigger specific events at a specific Time and/or on a specific Date.
If you do not specify Triggerdate it will execute the event on the specific Time every day.
If you do not specify Time it will execute on the specific Date.

Properties & Events :

AboutComponent	Component Information
CurrentTime	Current Time (Read-only)
CurrentDate	Current Date (Read-only)
TriggerDate	Date on which to Trigger
TriggerTime	Time at which to Trigger
CheckForTrigger	enables/disables triggering of the event
OnTrigger	the event that get triggered at a specific Time
OnNewTime	Executes everytime the time changes
OnNewDate	Executes everytime the date changes

TTrialRun

TTrialRun = Class(TComponent)

Use this Component to disable functions of your Application after a specific Date.

Procedures & Functions :

Function Execute : Boolean;

Properties & Events :

AboutComponent	Component Information
OnExpired	Event that executes on/after the expiry date
OnNotExpired	Event that executes before the expiry date
EndYear	Expiry Year
EndMonth	Expiry Month
EndDay	Expiry Day

TApplicationInfo

TApplicationInfo = Class(TComponent)

Properties & Events :

HintColor	Color of the Hints *do not use with THints
HintPause	Delay the Hint is visible *do not use with THints
ApplicationIcon	Icon of the Application
ShowHint	enables/disables Showing of Hints
ApplicationTitle	Title of your Application
OnActivate	Event that executes on Activation of your Application
OnDeactivate	Event that executes on Deactivation of your Application
OnException	Event that executes on an Exception
OnHelp	Event that executes on Help
OnIdle	Event that executes on Idle
OnMessage	Event that on a message that gets received
OnMinimize	Event on minimize
OnHint	Event on Hint
OnRestore	Event on restore
OnShow	Event on Show
AboutComponent	Component Information

TFileExecute

TFileExecute= Class(TComponent)

Runs,opens,explores,etc another Application from your Application.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
ExeName	Name of the Executable or filename of a specific file that belongs to an Application as specified in the Windows Settings
ExeParams	Parameters to execute
ExePath	Path where the Application executes
Wait	Wait flag for waiting until Program has terminated
WindowState	State of the Mainwindow of the Application
DsgnExecute	Execute in design Time
ExeStyle	Style in which to execute the Application

TQuickMail

TQuickmail = Class(TComponent)

Send a mail or a fax to another Person from your Application. This one has been successfully tested by many people around the World. I can say that since I have received E-mail from people not changing the E-mail address and testing it. The default E-mail was setup to be my own and in that style I received many empty E-Mails. Just do me a favor: Write something so I can at least have something to read.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
ToEmailOrFaxAddress	Address
ToName	Name of the Person you are addressing
TextToSend	Actual mail

Subject	Subject you wanting to write about
MailOrFax	Specify Email or fax
DsgnExecute	execute in designtime
ErrorNo	No of the error that occurred last
BeforeSend	Event that triggers before the send operation
OnError	On error event

TResourceInfo

TResourceInfo = Class(TComponent)

This Component holds Resource Information of your PC. All of these Resources get read in a separate Thread and refresh on any change. Attach the reading of the properties into Labels,Listboxes etc. in the OnNewValues event. Unlike TSystemInformation these values could change while your Application is running.

Properties & Events :

AboutComponent	Component Information
CurrentTime	Current Time
CurrentDate	Current Date
OnNewValues	Event for reading the values
CurrDriveSizeFree	Current free Drive Size
CurrDriveSize	Current max. Drive Size
CapsLock	Caps on/off
NumLock	Num on/off
ScrollLock	Scroll on/off
MemoryPercInUse	Memory in use
PhysMemFree	Physical Memory free
PhysMemTotal	Total Physical Memory on your PC
PageFileUsed	Used Pagefile size
VirtualMemUsed	Virtual Memory in use

TSystemInformation

TSystemInformation = Class(TComponent)

Properties & Events :

AboutComponent	Component Information
Organisation	Name of the Organization Windows is registered to
Owner	Name of the Person Windows is registered to
CPU	Type of CPU
OS	Type of Operating System
WinVersion	Version of Windows currently running
Build	Build No of Windows
TotalMemory	Total Memory that can be used
DriveC	Total Drive Size of your C-Drive
ProgramsFileDir	Directory where your Program Files are located
SystemRoot	Directory of your Windows/System Directory
UserName	Username currently in use
ComputerName	Name of the Computer
ScreenWidth	Width of the screen
ScreenHeight	Height of the screen

TSystemTray

TSystemTray = Class(TComponent)

This Component puts your Application on the System Tray when minimized.

Properties & Events :

Enabled	Enables/Disables the functionality
Hint	Hint of the Icon when on the Systemtray
PopupMenu	Popupmenu that shows on the SystemMenu
NormalIcon	Normal Icon on the tray
AboutComponent	Component Information
OnClick	
OnDblClick	
OnRightClick	
OnMinimize	
OnRestore	

TDigitalClock

TDigitalClock = Class(TCustomPanel)

Displays a digital Clock on the Form. If you change the font please resize the Clock in order to obtain the correct size. As you can see this Component has changed since one of my registered Users wanted real Digits inside the Watch. Nothing easier then that ... I simply took the coding out of the Iron Gamber - a game I wrote in the old Turbo Pascal Days. You won't believe how easy it was to integrate the object.

Properties & Events :

AboutComponent	Component Information
DigitShadowColor	Color of the Shadow of the Digit
DigitColor	Color of the Digits
BackColor	Color of the Background
Dialogframe	Shows a Dialogframe around the clock
DisplayType	Switch between Date and Time
BevelInner	Inner Frame
FrameBackground	Background in the frame
BevelOuter	Outer Frame

TWavePlayer

TWavePlayer = Class(TComponent)

Play a Wavefile from your Application. After seeing there are so many people who don’t know how to do it and are wanting to do that - here is the Component for Playing a Wave file from your Application.

Procedures & Functions :

Procedure Play(FileName: String; Async : Boolean);	plays an Individual Wave
Procedure Execute;	plays the configured Wave file

Properties & Events :

AboutComponent	Component Information
FileName	Filename of the Wave
Asynchronously	Type of Sound
OnError	Event on Error
DsgnExecute	Execute in Design Time

TFileCollector

TFileCollector = Class(TComponent)

Collect all file Locations fitting the specified Mask into a Stringlist.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
FileMask	the Filemask you want to apply (e.g. *.bmp will collect the locations of all Bitmaps in the Path)
InitialDirectory	Path from whereon you want to start to collect (e.g. C:\ will collect on your whole harddisk)
LocatedObjects	The Stringlist where found Locations are copied into
DsgnExecute	Execute your Search in Designtime

TFileOperator

TFileOperator = Class(TComponent)

The FileOperator can Copy,Move,Rename,Delete Files making use of Windows Explorer processing Screen.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
FilesFrom	Indicate the Filenames
DirectoryFrom	Indicate the Directory where the files are located
DirectoryToOrRenamedFilename	Indicate the Directory you are copying to or the new Filename when you rename a file
Operation	Indicate the type of Operation
OperationFlags	Indicate the Operationflags
DsgnExecute	Execute in design time
OnSuccessRename	
OnSuccessCopy	
OnSuccessDelete	
OnSuccessMove	
OnFailRename	
OnFailCopy	

OnFailDelete	
OnFailMove	

TTextFileView

TTextFileView = Class(TComponent)

A Replacement for notepad. Using this Component you will never have to test again if Notepad exists on the PC where your Application resides. Simply use it as your own Notepad.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
EditorCaption	Dialog Caption
EditColor	Color of the Edit
EditFonts	Font in the Edit
DsgnExecute	Execute in Design Time
MenuFile	Menu for File
MenuOptions	Menu for Options
MenuNew	Menu for New
MenuOpen	Menu for Open
MenuSave	Menu for Save
MenuPrint	Menu for Print
MenuExit	Menu for Exit
MenuWordWrap	Menu for WordWrap
MenuFont	Menu for Font
MenuBackground	Menu for Background
MenuDividers	Menu for Dividers
Menus	
OpenFileName	filename to open
OpenFileOnExecute	open file when executing
OnExecute	on executing
OnExit	on Exit
OnOpenFile	on opening a file
OnSaveFile	on saving a file
WindowState	State of the Window
WindowPosition	Position of the Window
BorderIcons	Icons in the Border
WindowWidth	Width of the Window
WindowHeight	Height of the Window
Icon	Icon of the Window
ReadOnly	Text is Read-only when set to true

TFileTextReplace

TFileTextReplace = Class(TComponent)

Scan through a file and do searches and replacements.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
FileFrom	File to Copy from
FileTo	File to Copy to
BeforeWrite	Before writing each individual String
CurrentString	The current String before writing

UseNoOfLines	Only Scan no of lines in the file
NoOfLinesOnly	Number of Lines
WriteThisLine	skip a specific line

TBackground

TBackground = Class(TComponent)

New looks, new feels - well Windows 2000 where are you ?
This component enables you to give your components almost any look and feel. Containing only the properties and procedures of the drawing effects   it is not usable by itself. If you change the properties in runtime please refresh all forms
containing components linking to the TBackground component.
The best way to operate this component is to put it onto your Mainform and link from all other forms to it.
To establish that add your Mainform in your Units implementation uses clause.

New ! Now even more Gradient Styles ... new Background Styles and its is possible to exchange any Color in your Texture with another one (eg. this could be clBtnFace for Buttons)

eg.

...
implementation
uses Mainform;
*{$R *.DFM}*
...

Properties & Events :

AboutComponent	Component Information
BackgroundStyle	bGradient - puts a gradient onto the Background bSolidcolor - uses a color bTexture - Uses a Bitmap (Image as Texture) bBrushFillSolidcolor - Uses SolidColor to fill the Background with the Brush indicated bBrushFillUseCurrent - Uses the Default Background of the Control and the indicated Brush
+BrushFill	BrushFill Settings
BrushStyle	Type of Brush to use
BrushColor	Color to use for the Brush
+Gradient	Gradient Settings
Begincolor	Color to begin with
Endcolor	Color to end with
Fillstyle	Style of Gradient
NoOfColors	Number of colors *less Colors = faster drawing
SolidColor	Color to use if theSetting is neither bTexture nor bGradient
+Texture	Texture Settings
Image	Image to use as Texture
ImageStyle	type of Display for the Image - isTile will tile your Image on the Background - isCenter will put one copy of your Image onto the Background - isStretched will stretch the Image across the Background
TransparentColor	uses one of the Colors Transparent if not set to clNone
TransparentBackground	the Color to use when any of the Colors in the Image is the TransparentColor
OnChanged	Event that triggers when the User changes any of the properties

TAboutComponent

The AboutComponent property which calculates all the Information about the Component.
This property will display a Dialog about every bit of Information on the Component and the

related release of the Component Collection it belongs to.

TFrame

New ! Now with even more styles to choose from ....

Properties & Events :

FrameStyle	Style of Frame
Framewidth	width of the Frame
Highlight	Color of the Highlight of the Frame
Shadow	Color of the Shadow of the Frame
Solid	Color for a solid frame
Offset	Specifies the Offset

TFont3D

Properties & Events :

Highlight	Color of the Highlight
Offset	Offset from the actual label
Repetition	Indicates a repetition from the Label
RepetitionColor	Color of the Repetition
Repetionwidth	Width of the Repetition
Shadow	Color of the Shadow
Style3D	Type of 3D
Threedimensional	enable/disable 3D Display

TButtonInfo

Properties & Events :

Caption	Caption of the button
ComfirmMessage	Message when button is pressed
ConfirmOnPress	Show a Confirmation Message
Enabled	Button state is enabled/disabled
Glyph	Glyph on the Button
Hint	Hint of the Button
ShowHint	Showhint of the Button
Visible	Button is Visible if set to true

TOffice97Btn

TOffice97Btn = Class(TGraphicControl)

The Microsoft Office 97 Button with a Background and Font3D property. Here is the new look and feel for your Applications.

Properties & Events :

AboutComponent	Component Information
Background	Background on the Component
Font3D	3D Font on the Button
Bordercolor	Color of the Border
LedIn	Color of the Led when the Button is pressed
LedNormal	Color of the Led when the Button is normal
Ledout	Color of the Led when the Button is selected
RoundSize	Make the Roundsize bigger and smaller
UnUsedColor	Please identify a Color that is not used on the Button for drawing Operations

TArrowBtn

TArrowBtn = Class(TGraphicControl)

The Arrow Button with a Background and Font3D property. Here is the new look and feel for your Applications. You can use it between Listboxes, as pointer … etc..

Properties & Events :

AboutComponent	Component Information
Background	Background on the Component
ButtonRaise	Height of the Bevel on the Button
Bitmap	Glyph for the Button
ButtonType	Direction the Button is pointing to
UnUsedColor	Please identify a Color that is not used on the Button for drawing Operations

TRoundBtn

TRoundBtn = Class(TGraphicControl)

The Round Button with a Background and Font3D property. Here is the new look and feel for your Applications. You can use it between Listboxes, as pointer … etc..

Properties & Events :

AboutComponent	Component Information
Background	Background on the Component
ButtonType	Direction the Button is pointing to
Bitmap	Glyph for the Button
UnUsedColor	Please identify a Color that is not used on the Button for drawing Operations

TMenuItemGlyph

TMenuItemGlyph = Class(TComponent)

Use this Component to attach a Glyph to any Item on any type of Menu.

Properties & Events :

AboutComponent	Component Information
CheckGlyph	Glyph on a checked Item
ItemGlyph	Glyph on a normal Item
MenuItem	The Item the glyph is attached to

TLookupEdit

TLookupEdit = Class(TMaskEdit)

A Editbox Component that allows you to click a button within to retrieve Data from any other Source.

Properties & Events :

AboutComponent	Component Information
ClickKey	Key which you press to Click the button
LookupButton	Button Properties
ReadOnlyClickable	If set to true then you can click the button even if the Editbox itself is marked read-only
OnButtonClick	Event that executes when the Button is clicked

TDBLookupEdit

TDBLookupEdit = Class(TMaskEdit)

A dataaware Version of the Editbox Component that allows you to click a button within to retrieve Data from any other Source.

Properties & Events :

AboutComponent	Component Information
ClickKey	Key which you press to Click the button
LookupButton	Button Properties
ReadOnlyClickable	If set to true then you can click the button even if the editbox itself is marked readonly
OnButtonClick	Event that executes when the Button is clicked

TDBMultiLabel

TDBMultiLabel = Class(TCustomLabel)

A Dataaware Version of the Label Component. The Background is drawn on the actual Font.
Note the color of the font must be a color not used in or under the Label !
The Heading in this Helpfile (AHM Component Collection) is designed using TMultilabel.

Properties & Events :

AboutComponent	Component Information
AsButton	Label becomes a button if set to true
FontBackground	Background on the Font
Background	Background of the Label
Font3D	Three-dimensional FontStyles
Frame	Frame around the Control

TRegEntry

TRegentry = Class(TComponent)

A Wrapper around the Registry. Use this Component to store Information about your Application in the Registry. The Name variable refers to the name of the entry and Value to the Value of the Identifier.

e.g. Name=Value

Procedures & Functions :

Function RStr(Name : String):String;	Read a String from the Area of the Registry
Procedure WStr(Name,Value : String);	Write a String to the Area of the Registry
Function RCryptStr(Name : String):String;	Read and decrypt a String from the IArea of the Registry (Passwords for example)
Procedure WCryptStr(Name,Value : String);	Write and encrypt a String to the Area of the Registry
Function Rint(Name : String):Integer;	Read a Integer from the Area of the Registry
Procedure WInt(Name : String;Value : Integer);	Write a Integer to the Area of the Registry
Function Rbool(Name : String;):Boolean;	Read a Boolean from the Area of the Registry
Procedure WBool(Name : String;Value : Boolean);	Write a Boolean to the Area of the Registry
Procedure Erase;	Erase the whole Section in the Area of the Registry

Properties & Events :

Area	Name of the Area in the registry
Section	Section in the Area of the Registry (if it doesn’t exist it will be created)
OnError	Event that occurs if one of the functions returns an error
AboutComponent	Component Information

TTimer32

TTimer32 = Class(TPersistent)

A new Timer Component. Creating a Thread makes your Application truly Multithreaded. I have used this new Timer in all Components making use of a Timer in some way or another.

Properties & Events :

AboutComponent	Component Information
Priority	Priority at which the thread is about to execute.

TExitWindows

TExitWindows = Class(TPersistent)

This Component is evetually back in my Collection. Its use lies within Installation programs and other Utilities that need to reboot/shutdown the machine. It can also be used to as a nice addon for Applications that run off straight from the Taskbar as Menu. Please just don’t try it out in the Development Environment without saving your Project first ... it does reboot your machine afterall.

Procedures & Functions :

Procedure Execute;

Properties & Events :

AboutComponent	Component Information
ExitType	Type of Exit from Windows
Confirm	Confirm before running the command
ConfirmMessage	Message for the Confirmation

TOnlyOne

TOnlyOne = Class(TComponent)

This Component is evetually back in my Collection. I found the Code for it distributed on the Internet and found people looking for it. So what do I do ? I decided to put it in. Its not charged as so many other new Components in this release since the price will not increase for it. If you need the code just send me some E-Mail and I will send it back to you or tell you where to find it .

To use it just simply drop it on and set the property indicating how many instances of your Software you want to enable. Then indicate what you want to do when this number is exceeded.

Properties & Events :

AboutComponent	Component Information
AllowedInstances	Number of Instances to allow
ActionOnExceeded	Indicates on what Actions to do when the number of Instances is exceeded
OnInstanceExceeded	You own actions you want to add

I am presently working for Paradigm Systems Technology as a Application Systems Developer. I am holding a Work Permit in the Republic of South Africa which expires every ½ a year. An Application for Permanent Residence has not been accepted yet. I would be accepting Jobs in and outside of South Africa if my requirements would be met and the field of Development is in Delphi. For further Info contact me via e-mail.

Thank you, Alexander Mehlhorn

Name :	Alexander Henning Mehlhorn
Parents :	Prof. Dr. Hans-Georg Mehlhorn Prof. Dr. sc. Gerlinde Mehlhorn
Residential Address :	Barlinka 21 Van Willigh Street Centurion 0046 South Africa
Postal Address :	P.O. Box 11037 Centurion 0046 South Africa
Home Tel :	(012) 663 5806
Date of Birth :	11-July-1973
Age :	23
Nationality :	German
Marital Status :	Single
Home Language :	German *English for the past 5 years
Other Languages :	English, Afrikaans
Present Health :	Very good
Education :	Germany Std. 1 -10 , Abitur (Std. 11) South Africa Matric still busy studying Bachelor of Science at the University of South Africa (UNISA)

Career History

Company :	DK Instruments
Position Held :	Working on Computers and Specialised Instruments
Employment Dates :	1.April 1994 - 30.September 1994
Salary :	XXXXX
Duties :	Maintenance of Equipment Maintenance of Software Development of Software
Reason for Leaving :	Employer moved Overseas
Company :	Paradigm Systems Technology
Position Held :	Tester Junior Application Systems Developer Baseline Manager Application Systems Developer specialised in 3GL and 4GL
Employment Dates :	1.October.1994 -
Salary :	XXXXX
Duties :	Testing of Software Fixing of corrupted Software Inputs to Documentation for Development of Applications Software Reference Manuals Correction of Application Software Unit Tests Development on Paradigms Configuration Management System

Development on the Integrated Package of

Configuration Management System

Maintenance Management System

Material Management System

Acquisition Management System

Correction of 3rd Line Support to Clients

Inputs to System Engineering for Development of Application Software Data Model and Data Dictionary

Porting of Applications to new Platforms

Maintenance of the Baselines

Maintenance of the Releases

Development of Utilities to help Developers

Fix and correct Problems

Development of Tools for the Development Process

Setting up of Environments for Developers/Testers

Development of specialist Software for the Documentation Process

Development of Installation Programs for Paradigm’s Software

Development of Porting/Translation Program from Assyst (under OS2) to Uniface(under Windows)

Platform Skills : OS2,Windows (all releases - NT,95,3.1,Workgroups) ,Dos,Unix

Language Skills : Basic,Visual Basic, Java, Assyst, Pascal, C, **Delphi**, Uniface, SQL

Database Skills : Dbase, Paradox, Interbase, Oracle

for more Info please contact me via e-mail

AHM Component Collection VEnhanced97

What’s New

This release is somewhat different then my others. The first thing you might have noticed is that it is considered to be a Beta for Delphi 3.0 Users and a Final for Delphi 2.0 Users. Well due to the differences in Delphi and me not developing such a huge release for multiple Delphi’s (I am only one Person with once PC ... running a couple of Delphi’s is not my Idea of programming even thought I absolutly love Delphi) I am now dropping the Delphi 2.0 support. I can not do with it any longer. If there is someone out there who wants to help - well give me a shout and I will contact you and I am sure we can work something out.

Second of all. I have been trying to get hold of my life for quiet some time now as life is running away from me. I am going to Germany and to America during the course of August to sort out my life and to see what people have to offer. This is mainly planned as a holiday but I will be receiving e-mail on a maybe not to regular basis. Please be patient during August on replies about buying the collection. It might take a day or two longer to get an reply but I will be forwarding my mail whereever I will be. (Thanks to the Internet once again).

The Beta of the Delphi 3.0 Version will hopefully soon afterwards be released as the Final. I am just planning to finish a couple of things that I had started. To the Delphi 2.0 Users I thank you and wish you all the best in developing further as I hope to see you on Delphi 3.0 soon.

What’s Old

Besides Delphi 2.0 and AHM 97.32 nothing much ....
Sorry I couldn’t help that as I have nothing to add this time around. Delphi is still the same and I can not say that something major has changed on a basis that would stamp things old. TExecProgram is now TFileExecute for reasons of useability but thats about it.

Disclaimer

The Author cannot and does not warrant that any functions contained in the Software will meet your requirements, or that its operations will be error free. The entire risk as to the Software performance or quality, or both, is solely with the user and not the Author. You assume responsibility for the selection of the component to achieve your intended results, and for the installation, use, and results obtained from the Software.

The Author makes no warranty, either implied or expressed, including without limitation any warranty with respect to this Software documented here, its quality, performance, or fitness for a particular purpose. In no event shall the Author be liable to you for damages, whether direct or indirect, incidental, special, or consequential arising out the use of or any defect in the Software, even if the Author has been advised of the possibility of such damages, or for any claim by any other party.

All other warranties of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose, are expressly excluded.