



Product Documentation

ToGo Edition Quick Start

InterBase XE

September, 2011

© 2011 Embarcadero Technologies, Inc. Embarcadero, the Embarcadero Technologies logos, and all other Embarcadero Technologies product or service names are trademarks or registered trademarks of Embarcadero Technologies, Inc. All other trademarks are property of their respective owners.

Embarcadero Technologies, Inc. is a leading provider of award-winning tools for application developers and database professionals so they can design systems right, build them faster and run them better, regardless of their platform or programming language. Ninety of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero products to increase productivity, reduce costs, simplify change management and compliance and accelerate innovation. The company's flagship tools include: Embarcadero® Change Manager™, RAD Studio, DBArtisan®, Delphi®, ER/Studio®, JBuilder® and Rapid SQL®. Founded in 1993, Embarcadero is headquartered in San Francisco, with offices located around the world. Embarcadero is online at www.embarcadero.com.

September 7, 2011

- Introduction..... 1
 - Comparing InterBase ToGo and InterBase Desktop 1
 - System Requirements..... 2
 - ToGo Edition Users: 2
- Using the ToGo Edition 3
 - Windows Platforms..... 3
 - Developing with the ToGo Edition for Windows3
 - Deploying with the ToGo Edition for Windows4
 - Mac OS X Platform..... 6
 - Developing with the ToGo Edition for Mac6
 - Deploying with the ToGo Edition for Mac7
- Licensing ToGo..... 9
 - The Default Folder for ToGo..... 10
 - Specify a Temp Directory..... 10
- Developing Applications for InterBase ToGo 10
 - Using IBConsole with InterBase ToGo 11
- ToGo Requirements and Constraints 11

INTRODUCTION

The InterBase ToGo database engine can be embedded in applications by directly using the InterBase database engine library. It does not have to be installed on any server or end-user workstation and it does not require that InterBase be installed on a machine before use like InterBase Server and Desktop.

ToGo has no separate database server process running, and the database engine executes in the application process space. The embedded InterBase engine runs faster with these types of database applications because network connections are not used and data is not passed between the application and a separate database server. This gives application developers more choice and flexibility in their database design and deployment. Some applications for the embedded service include small devices and public kiosks, and the lowest end version of Value Added Resellers (VAR) applications which are built using InterBase.

For information on becoming a Value Added Retailer see <http://www.embarcadero.com/products/interbase/faq#oem>

Only one application is allowed to connect to the database; the application exclusively locks the database file. However this application is allowed to make multiple connections to the database, so a multi-threaded application can make a single connection or multiple connections and access the database simultaneously. This also means that no other external client can access the database when it is being used (connected to) by the application using the embedded database engine.

COMPARING INTERBASE TOGO AND INTERBASE DESKTOP

- 1 InterBase ToGo runs as an embedded process. InterBase Desktop requires you to start the engine as a service or an application. This makes deployment easier.
- 2 InterBase ToGo requires no installation. You just copy files into the application directory.
- 3 Both editions require you to become a VAR if you wish to deploy to your customers.
- 4 InterBase ToGo allows only one application to use a database at a time. InterBase Desktop application allows multiple applications to use a database at the same time. A common example of multiple applications using the same database is a Delphi application and a reporting application both being run at the same time against the same database.
- 5 Both editions use only local databases. However, both can be used to access databases on remote machines as long as the remote machine has a Server Edition of InterBase running.
- 6 Neither edition can query system temporary tables.
- 7 Performance monitoring via IBConsole is disabled in both editions.
- 8 Multiple applications, even though they might be using different databases, cannot use the same InterBase ToGo license at the same time. Each application requires separate licensing. InterBase Desktop does not have this restriction.
- 9 It is important to note that ToGo does not support "shadowing" and "online dumps".

SYSTEM REQUIREMENTS

The InterBase ToGo edition is available on the following Windows platforms and can access any InterBase database created by InterBase desktop or server editions, versions 2007 and later.

- Microsoft Windows Vista
- Microsoft Windows 7 (32-bit and 64-bit)
- Microsoft Windows XP (SP2)
- Microsoft Windows Server 2003, 2008
- Microsoft Windows Server 2008 R2 (64-bit)
- Microsoft Windows 2000 (SP4)

InterBase ToGo is now also available on Apple MAC OS X.

TOGO EDITION USERS:

VARs and OEMs can install and embed the InterBase database service within their applications. They can now include InterBase within their applications by linking in with the new ToGo edition library. Also, a VAR can access an InterBase database from the CD or DVD without the need to install anything on the machine. This means InterBase does not access anything from the Windows Registry. There is no need for registry changes because all needed settings are provided by options in the InterBase configuration file.

The InterBase log files, lock files, and event files now go to a [specified temporary directory](#) on the machine as opposed to the install directory, as the install directory can now be on a read-only media. The ToGo edition can be used in conjunction with Embedded User Authentication (EUA), to control user access to a database.

Application Developers can link their applications with an included library containing the InterBase database service so nothing is needed other than this library to access InterBase database files. Once the library is included there is no need to start the InterBase database service, or to stop the service once the application exits, as the database service is in process.

The database file is compatible across all the editions of InterBase, so a VAR can upgrade a particular customer installation to the Desktop or Server edition with minimal changes to his application.

For more information on developing applications see: [Developing Applications for InterBase ToGo](#).

USING THE TOGO EDITION

WINDOWS PLATFORMS

The file `ibtogo_min.zip` is the minimum file set required to deploy InterBase ToGo. This file set is primarily intended for VARs (Value Added Retailers) deployment.

The `ibtogo.zip` file includes the full file set for InterBase ToGo, and can be used for development with the ToGo edition.

- 1 You can unzip these two files into a directory such as `c:\Interbase\ToGo_Min` and `c:\Interbase\ToGo_full` so you can copy them later.
- 2 Depending on which type you want to deploy (minimum or full file set), copy the required InterBase config and `interbase.msg` file in an `interbase` sub-directory under your application directory.

The location of the InterBase configuration file and other files such as `interbase.msg` and `license` sub-directory are limited to the "interbase" sub-directory relative to the application. So if the application's running directory is `<app_dir>` then the configuration file location is always restricted to "`<app_dir>\interbase`". The `INTERBASE` environment variable is ignored.

NOTE: It is recommended that you use EUA which means that you will not need the `admin.ib`. If you do not want to use EUA, copy your `admin.ib` to the `interbase` sub-directory.

DEVELOPING WITH THE TOGO EDITION FOR WINDOWS

- 1 In order to develop your application to work with the ToGo edition, replace the "`gds32.lib`" on your link path with the following files:
 - For CodeGear-based linkers the file is located in the `<app_dir>\interbase\sdk\lib\ibtogo.lib`
 - For Microsoft-based linkers the file is located in the `<app_dir>\interbase\sdk\lib_ms\ibtogo_ms.lib`
- 2 If you want to use the ToGo edition and `ibtogo.dll` with older applications or applications linked with `gds32.lib`, you need to copy the `ibtogo.dll` to `gds32.dll` and make sure this is the first in your application's path. `ibtogo.lib` and `gds32.lib` both surface and support the same API so they can be moved from one to another.

NOTE: If you are developing with the ToGo edition you will need the complete ToGo Edition set (The `ibtogo.zip` file includes the full file set for InterBase ToGo).

DEPLOYING WITH THE TOGO EDITION FOR WINDOWS

- 1 When deploying with the ToGo edition, make sure the `ibtogo.dll` is available to your application when it is launched. The easiest way to ensure this is to include it in the same directory as your application.

The DLL is a file specially created to contain the function code and the definitions; a DLL is used by linking in a `.lib` file. An application linked using the DLL needs the DLL to be present at run time on the deployment machine, and be in the search PATH of the application.

- 2 Also create a InterBase sub-directory under the directory where "`ibtogo.dll`" is located (if you followed the recommendation above this should be your application directory) and copy the required InterBase config and `interbase.msg` file. InterBase licenses need to be made available in the InterBase/license folder. It is recommended that you use EUA which means that you will not need the `admin.ib`. If you do not want to use EUA copy your `admin.ib` to the interbase sub-directory.
- 3 You might need the internationalization, UDF, and other InterBase provided libraries if you use them in your application.
- 4 For ToGo deployment you just need the runtime fileset defined in the following table; you do not need to deploy with the full fileset.

File List for Windows 32-bit and 64-bit ToGo Editions:

File Name	32-Bit	64-Bit	Min. (runtime) kit	Full Kit	Notes
<code>ibtogo.dll</code>	Yes		Yes	Yes	Can be located either in the <run directory> or the application directory or anywhere in one of the directories included in the PATH
<code>ibtogo64.dll</code>		Yes	Yes	Yes	Can be located either in the <run directory> or the application directory or anywhere in one of the directories included in the PATH
<code>gbak.exe</code>	Yes	Yes		Yes	
<code>gfix.exe</code>	Yes	Yes		Yes	
<code>gpre.exe</code>	Yes	Yes		Yes	
<code>gsec.exe</code>	Yes	Yes		Yes	
<code>gstat.exe</code>	Yes	Yes		Yes	
<code>isql.exe</code>	Yes	Yes		Yes	
<code>InterBase\interbase.msg</code>	Yes	Yes		Yes	
<code>InterBase\ibconfig</code>	Yes	Yes		Yes	

File Name	32-Bit	64-Bit	Min. (runtime) kit	Full Kit	Notes
InterBase\admin.ib	Yes	Yes	Yes	Yes	Not needed if you are using embedded user authentication, which is recommended.
InterBase\License.txt	Yes	Yes	Yes	Yes	
InterBase\loss_license_notice.txt	Yes	Yes	Yes	Yes	
InterBase\intl\gdsintl.dll	Yes	Yes	Yes	Yes	
ib_util.dll	Yes			Yes	
ibxml.dll	Yes			Yes	
ib_util64.dll		Yes		Yes	
ib_xml64.dll		Yes		Yes	
InterBase\bin\LicenseManager.exe	Yes	Yes		Yes	
InterBase\bin\LicenseReg.exe	Yes	Yes		Yes	
InterBase\bin\sanctuarylib.dll	Yes	Yes		Yes	
InterBase\license\ibxetogo.slip	Yes	Yes	Yes	Yes	InterBase license file
InterBase\SDK\lib_ms\ibtogo_ms.lib	Yes			Yes	
InterBase\SDK\lib_ms\ib_util_ms.lib	Yes			Yes	
InterBase\SDK\lib_ms\ibxml_ms.lib	Yes			Yes	
InterBase\SDK\lib_ms\ibtogo64_ms.lib		Yes		Yes	
InterBase\SDK\lib_ms\ib_util64_ms.lib		Yes		Yes	
InterBase\SDK\lib_ms\ibxml64_ms.lib		Yes		Yes	
InterBase\SDK\lib\ibtogo.lib	Yes			Yes	
InterBase\SDK\lib\ib_util.lib	Yes			Yes	
InterBase\SDK\lib\ibxml.lib	Yes			Yes	
InterBase\SDK\include\ibase.h	Yes	Yes		Yes	
InterBase\SDK\include\iberror.h	Yes	Yes		Yes	
InterBase\SDK\include\ibxml.h	Yes	Yes		Yes	
InterBase\SDK\include\ibxml_prot.h	Yes	Yes		Yes	
InterBase\UDF\ib_udf.dll	Yes	Yes		Yes	
InterBase\examples\database\employee.gbk	Yes	Yes		Yes	
InterBase\examples\database\employee.gdb	Yes	Yes		Yes	

File Name	32-Bit	64-Bit	Min. (runtime) kit	Full Kit	Notes
InterBase\examples\database\intl emp.gbk	Yes	Yes		Yes	
InterBase\examples\database\intl emp.gdb	Yes	Yes		Yes	
InterBase\Doc\APIGuide.pdf	Yes	Yes		Yes	
InterBase\Doc\DataDef.pdf	Yes	Yes		Yes	
InterBase\Doc\DevGuide.pdf	Yes	Yes		Yes	
InterBase\Doc\EmbedSQL.pdf	Yes	Yes		Yes	
InterBase\Doc\LangRef.pdf	Yes	Yes		Yes	
InterBase\Doc\OpGuide.pdf	Yes	Yes		Yes	
InterBase\Doc\ReleaseNotes.pdf	Yes	Yes		Yes	

MAC OS X PLATFORM

DEVELOPING WITH THE TOGO EDITION FOR MAC

In order to develop your application to work with ToGo edition, replace the InterBase client library referred to by the "libgds.dylib" or "-lgds" on your link path (DYLD_LIBRARY_PATH) with "libibtogo.dylib" or "-libtogo", which refers to the new ToGo library.

The ToGo library, libibtogo.dylib, is located in the top-level directory of the ToGo zip file. If you want to use the ToGo edition and libibtogo.dylib with older applications linked with libgds.dylib for Server Edition, you have two options. Your choice will depend on what your deployment goal is.

In the first case where you have an executable "ibapp" (which uses libgds.dylib) but now want to create a new executable targeting the ToGo edition. You need to follow these steps on a Mac OS X Terminal window.

- 1 Copy ibapp to ibapp_togo.
- 2 Use the **install_name_tool** command to alter the InterBase library your application needs; use **otool** command to verify the change.
 - `install_name_tool \-change libgds.dylib libibtogo.dylib ibapp_togo`
 - `otool \-L ibapp_togo`

In the second case where you want to just have the same application switch to using the libibtogo.dylib library as libgds.dylib, you can do the following.

- 1 Copy libibtogo.dylib as libgds.dylib.
- 2 Use the **install_name_tool** with `\-id` option to change the install name of the dynamic library. The following command instructs the dynamic loader to recognize libibtogo.dylib as libgds.dylib for loading by applications. As always, ensure your DYLD_LIBRARY_PATH is set accordingly.
 - `install_name_tool -id libgds.dylib libgds.dylib "`

DEPLOYING WITH THE TOGO EDITION FOR MAC

When deploying with the ToGo edition, you just need to make sure that libibtogo.dylib is installed and the environment variable DYLD_LIBRARY_PATH has been updated to include the location of the libibtogo.dylib

You also need to create an **interbase** sub-directory under your application directory and copy the required "ibconfig" and interbase.msg file. The license sub-directory must also be located here.

The location of the InterBase configuration file and other files such as interbase.msg are limited to the "interbase" sub-directory relative to the application. The interbase.log file also defaults to the interbase sub-directory on Mac OS X. You can control the location of the interbase.log file by setting the following environment variable, in order of precedence.

```
# INTERBASE_TMP environment variable  
# TMP environment variable.
```

It is recommended you use EUA which means that you do not need the admin.ib. If you do not want to use EUA, copy your admin.ib to the interbase sub-directory.

You might need the i18n (gdsintl), UDF (ib_udf) and other InterBase provided libraries if you use them in your application. Please update your configuration file, ibconfig, to include the location of the UDF directory if it is not the default interbase/UDF.

For ToGo deployment you just need the runtime fileset mentioned below; you can choose to deploy the full set should you need to use the InterBase command-line tools and documentation.

File List for Mac OS X 32-bit ToGo Edition:

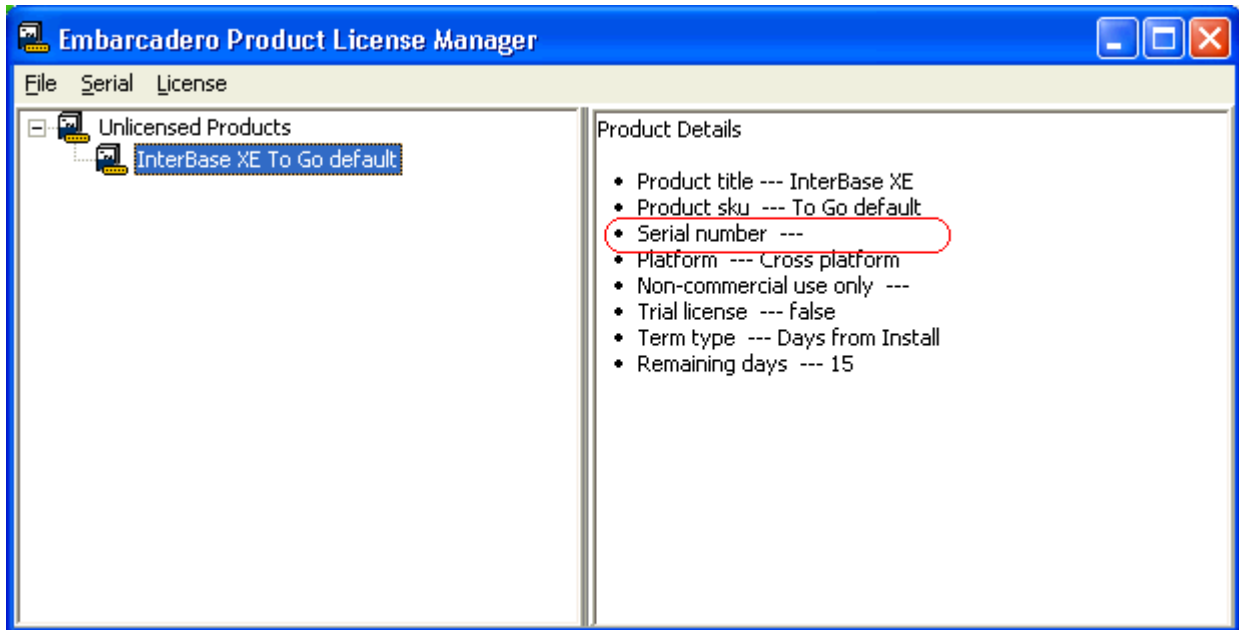
File Name	Min. (runtime) kit	Full Kit	Notes
libibtogo.dylib	Yes	Yes	Can be located either in the <run directory> or the application directory. The location must be included in DYLD_LIBRARY_PATH environment variable.
gbak		Yes	
gfix		Yes	
gpre		Yes	
gsec		Yes	
gstat		Yes	
isql		Yes	
interbase\interbase.msg	Yes	Yes	
interbase\ibconfig		Yes	
interbase\admin.ib	Yes	Yes	Not needed if you are using Embedded User Authentication (EUA), which is recommended
interbase\license.txt	Yes	Yes	
interbase\oss_license_notice.txt	Yes	Yes	
interbase\readme.html		Yes	
interbase\registration_FAQ.html		Yes	
interbase\IBsetup.html		Yes	
interbase\intl\gdsintl	Yes	Yes	
libib_util.dylib		Yes	
libibxml.dylib		Yes	
interbase\bin\LicenseManagerLauncher		Yes	
interbase\bin\lmlauncher.jar		Yes	
interbase\bin\cglm.ini		Yes	
interbase\license\ibxetogo.slip	Yes	Yes	InterBase license file
interbase\include\ibase.h		Yes	
interbase\include\iberror.h		Yes	
interbase\include\ibxml.h		Yes	
interbase\include\ibxml_proto.h		Yes	
interbase\UDF\ib_udf		Yes	
interbase\examples\employee.gbk		Yes	
interbase\examples\employee.gdb		Yes	
interbase\examples\intlemp.gbk		Yes	

File Name	Min. (runtime) kit	Full Kit	Notes
interbase\examples\intlemp.gdb		Yes	
interbase\Doc\APIGuide.pdf		Yes	
interbase\Doc\DataDef.pdf		Yes	
interbase\Doc\DevGuide.pdf		Yes	
interbase\Doc\EmbedSQL.pdf		Yes	
interbase\Doc\LangRef.pdf		Yes	
interbase\Doc\OpGuide.pdf		Yes	
interbase\Doc\ReleaseNotes.pdf		Yes	

LICENSING TOGO

If you are an InterBase VAR and have received a superkey slip license file for distribution, please deliver the license file into the <interbase>/license folder. The following steps are only required if you want to register an individual S/N.

- 1 To license InterBase ToGo, execute the License Manager tool (licensemanager.exe on Windows, LicenseManagerLauncher on Mac OS X) from the bin folder. This opens the Embarcadero Product License Manager dialog.
- 2 When you downloaded either a trial version or purchase a licensed version, you will receive an email with a serial number. You can then copy this serial number into License Manager dialog.



- 3 All a VAR must do is copy their .slip file into the license folder and they have all the files they need.

THE DEFAULT FOLDER FOR TOGO

The location of the InterBase configuration file and other files such as `interbase.msg` are limited to the "interbase" sub-directory relative to the application. The `interbase.log` file also defaults to the interbase sub-directory on Mac OS X. You can control the location of the `interbase.log` file by setting the following environment variable, in order of precedence.

- 1 `INTERBASE_TMP` environment variable
- 2 `TMP` environment variable.

SPECIFY A TEMP DIRECTORY

In order to allow read-only media to access and create temporary files, InterBase relies on a configuration parameter called "`INTERBASE_TMP`". The "`INTERBASE_TMP`" controls location of write files such as "`interbase.log`". If the "`INTERBASE_TMP`" configuration parameter is not set, then the following order is used to check for the existence of environment variables and use the first one found:

- The path specified by the `TMP` environment variable.
- The path specified by the `TEMP` environment variable.
- The path specified by the `USERPROFILE` environment variable.
- The Windows directory.

For more information on specifying a Temp Directory see: *Configuring Sort Files* in the **Operations Guide**.

DEVELOPING APPLICATIONS FOR INTERBASE TOGO

The simplest way to develop an application for InterBase ToGo is to do development using InterBase Developer edition. The InterBase Developer Edition is included with Embarcadero IDE's and can be downloaded for free from Embarcadero's web site.

When you are ready to use your application with InterBase ToGo, all you need to do is copy your executable to the same directory as `ibtogo.dll`. If you use this strategy for development you do not need to worry about paths. Also, if you use this strategy you will not need to worry whether any other installations of InterBase are on the same computer.

InterBase ToGo gets loaded using the Windows standard search order. This is important information if you elect to not place your executable in the same directory as `ibtogo.dll`. If you are unfamiliar with the Windows standard search order, it is strongly recommended that you read about it [here](#).

NOTE: The interface to InterBase ToGo is via `ibtogo.dll`. This is different than InterBase Server/Desktop editions which uses `gds32.dll`. However, `ibtogo.dll` and `gds32.dll` have the same interface. This means applications written for InterBase Desktop edition and 98% of all applications written for InterBase Server edition work unmodified by simply renaming `ibtogo.dll` to `gds32.dll`.

USING IBCONSOLE WITH INTERBASE TOGO

While IBConsole is not included with InterBase ToGo, it works fine with InterBase ToGo if you require a graphical user interface tool to manage your database. IBConsole may be obtained from the InterBase Developer Edition. Depending on the version of IBConsole you are using, you may need to rename/copy `ibtogo.dll` to `gds32.dll` as noted above. Keep in mind that InterBase ToGo does not allow multiple applications to use the same database at the same time, so you cannot use IBConsole with your database while your application is using it at the same time.

TOGO REQUIREMENTS AND CONSTRAINTS

- Listed below is other functionality not available on the ToGo edition:
 - performance monitoring
 - database point in time and disaster recovery
 - database linger
 - shadowing
 - on-line dumps.
- If a function is not present in the Desktop edition, it is also not present in the ToGo Edition.
- No Installer, IBManager, or IBConsole are provided for this edition of InterBase. Installer and IBManager are not needed for ToGo.
- No JDBC access is available for the ToGo edition.
- If you need access to the above features, you can upgrade to Desktop or Server Edition(s). This is on the same lines as the restrictions we have placed with performance monitoring, where only the server edition can access this feature.