



Hit the Ground Running with InterBase

What you need to know about developing, deploying
and managing your InterBase instances





Agenda

Getting IB Up
and Running

Developing and Deploying
InterBase Applications

Server

Mobile

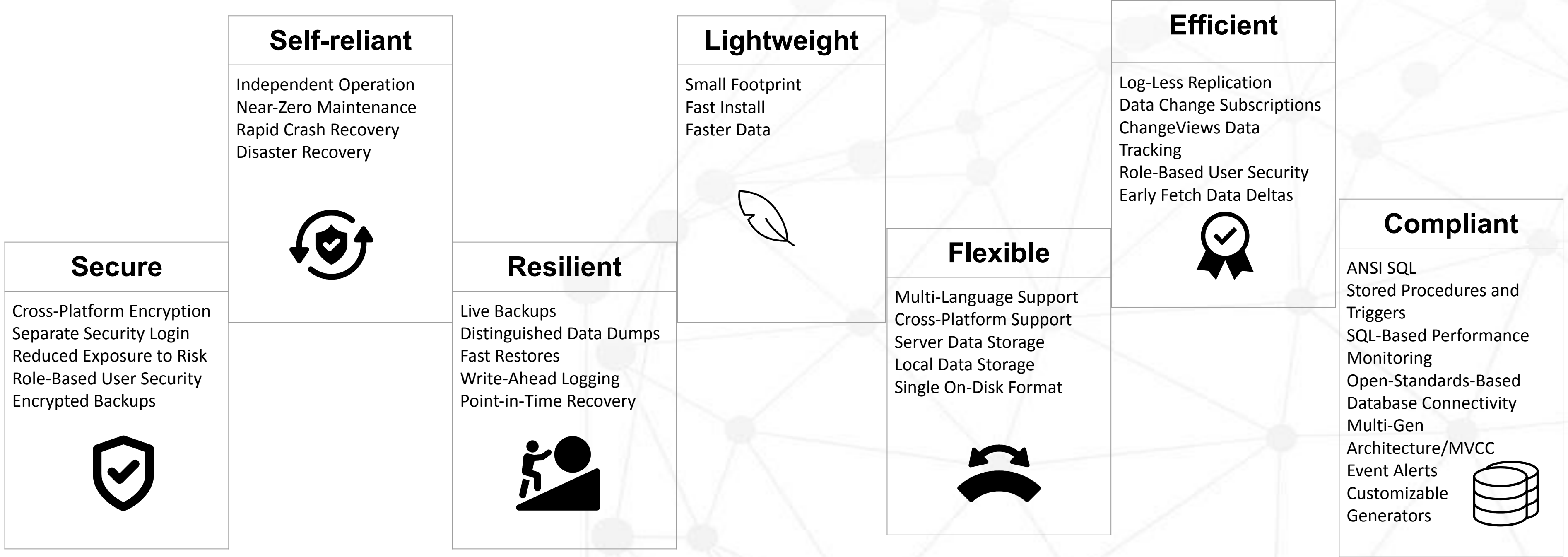
Managing InterBase





What is InterBase?

InterBase® is a **full-featured, high performance, ultrafast**, encryptable, scalable, relational and embeddable multi-platform SQL database with commercial-grade **data security, disaster recovery** and **change synchronization** for developers who are looking to embed a **low-cost, zero-admin, secure** database into their cross-platform connected applications. Global companies across a range of industries have a competitive advantage with InterBase® through advanced **data protection, security, speed** and **performance**.





InterBase Versions

InterBase Server

1 to 100's of network users

Fully featured + Server monitoring



InterBase Desktop

Local Instance

Fully featured



InterBase ToGo

Deeply Embedded

Fully featured
Optimized for single app use



IBLite

Deeply Embedded

100MB file limit, No Encryption
Optimized for single app use



Developer Edition

20 users max.

Restart required





Getting InterBase Up and Running





How to get InterBase

1. RAD Studio

- a. Developer Edition
- b. IBLite/ToGo (mobile only) included

2. Embarcadero Website

- a. Developer Edition
- b. Server Edition Trial
- c. ToGo Edition Trial









Sample databases

RAD Studio

C:\Users\Public\Documents\Embarcadero\Studio\21.0\Samples\Data

-  dbdemos.gdb
-  EMPLOYEE.GDB
-  EVENTS.GDB
-  MASTSQL.GDB

InterBase

C:\ProgramData\Embarcadero\InterBase\InstanceName\examples\database

-  employee.gbk
-  employee.gdb
-  intlemp.gbk
-  intlemp.gdb





Developing and Deploying Applications with InterBase



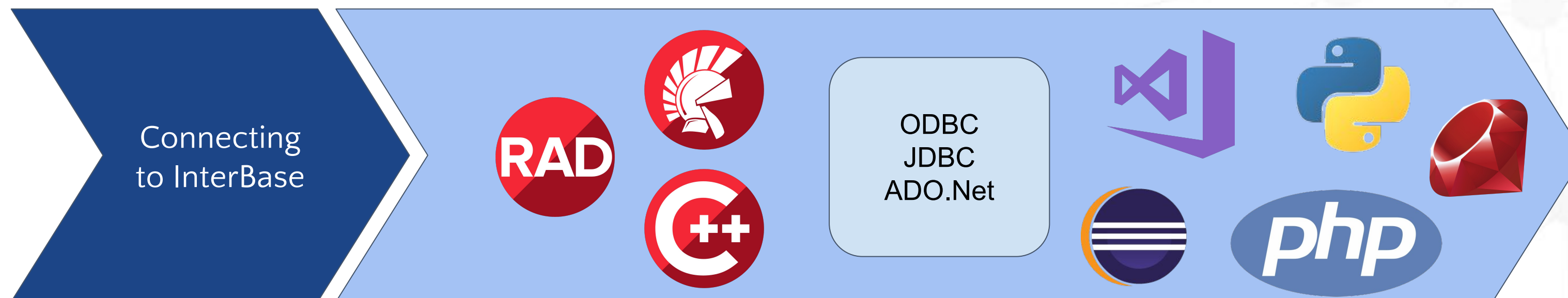


IBConsole



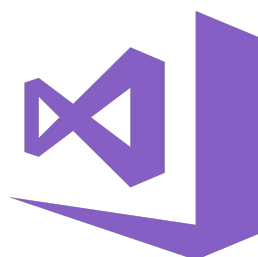


Connecting to InterBase



FireDAC

The official InterBase documentation is online at <http://docwiki.embarcadero.com/>



GitHub : <https://github.com/Embarcadero/IB.NETDataProvider>

NuGet Drivers : <https://www.nuget.org/packages/InterBaseSql.Data.InterBaseClient/>

Entity Framework : <https://www.nuget.org/packages/InterBaseSql.EntityFrameworkCore.InterBase/>

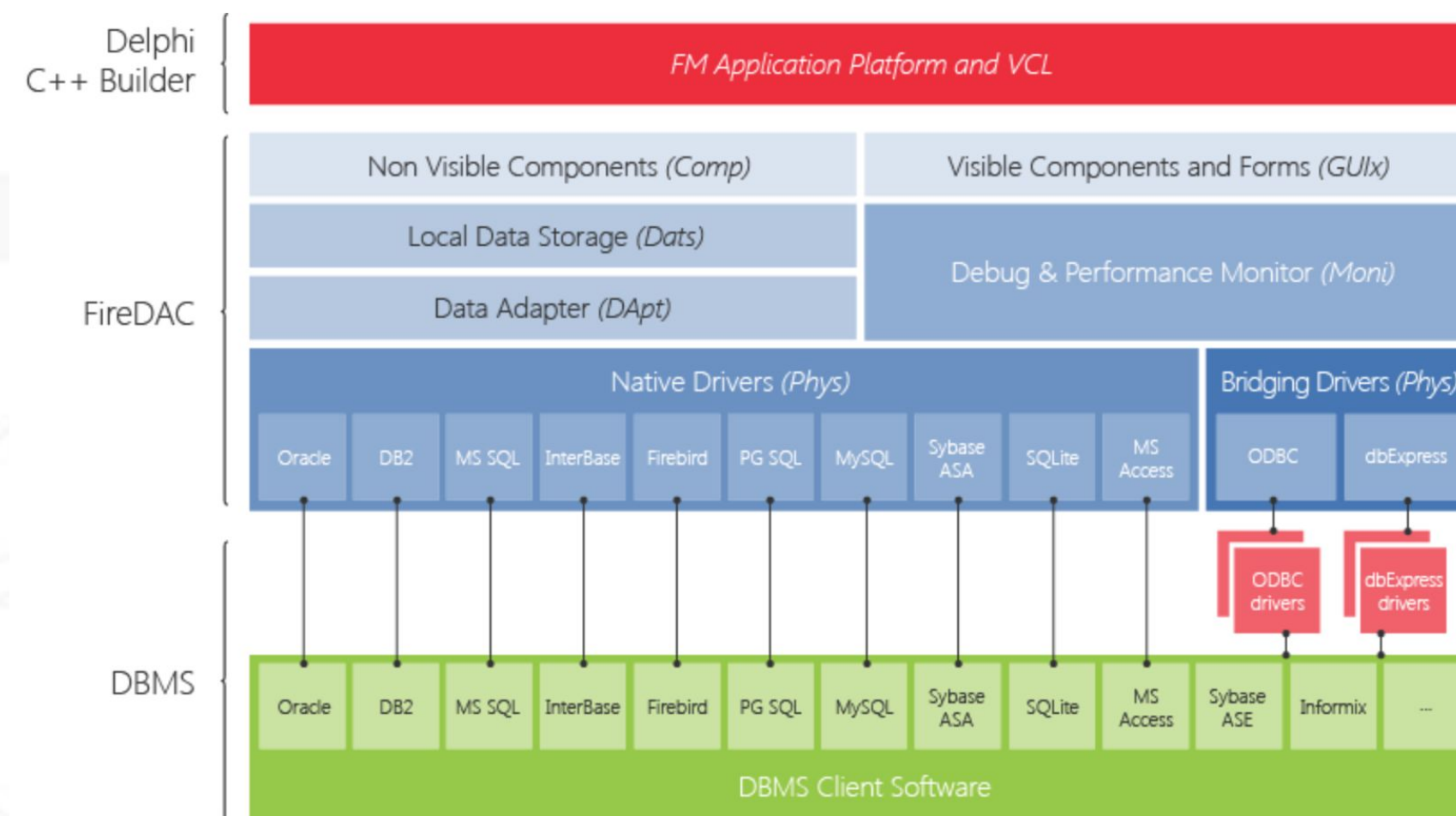




FireDAC

- Is a unique set of Universal Data Access components for developing VCL and FMX applications for Delphi, C++Builder.
- Enables native high-speed direct access to all of your database from Delphi or C++Builder.
- InterBase, Oracle, SQLite, MSSQL, MySQL, PG SQL and more.

Universal Enterprise Data Connectivity





Change Views

“Subscription based” model used to identify what data has been changed in the database.

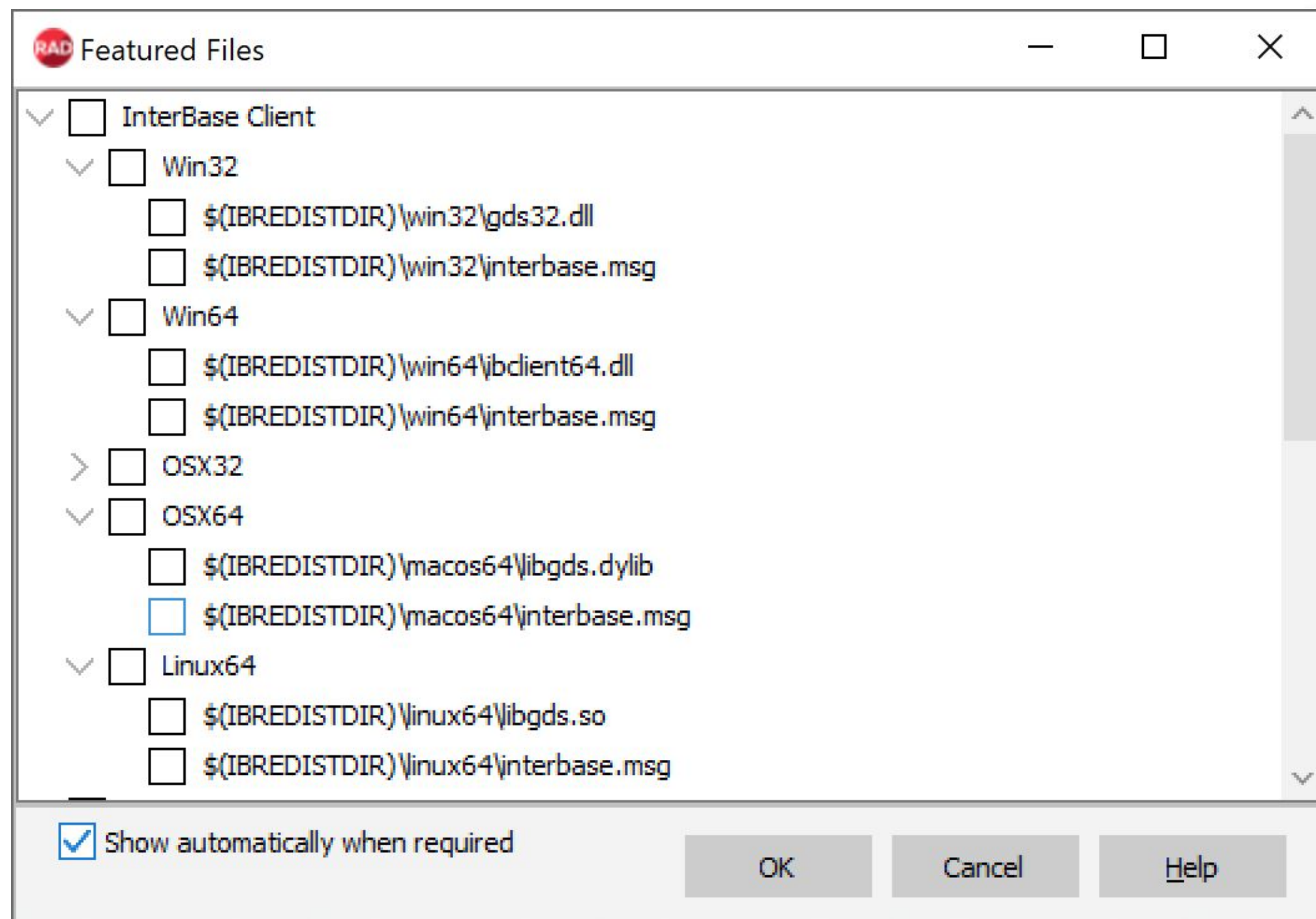
- Subscription can run:
 - During a connection
 - Spanning a connection
- Each subscriber can have multiple devices/subscriptions to the same data.





Desktop Deployment

Project > Deployment > Add Featured Files >





ToGo Deployment

InterBase ToGo

Win32

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\win32_togo\ibtogo.dll
- ☒ \$(IBREDISTDIR)\win32_togo\ibxml.dll
- ☒ \$(IBREDISTDIR)\win32_togo\ib_util.dll
- ☒ \$(IBREDISTDIR)\win32_togo\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\win32_togo\interbase\ADMIN.IB
- ☒ \$(IBREDISTDIR)\win32_togo\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\win32_togo\interbase\license.txt
- ☒ \$(IBREDISTDIR)\win32_togo\interbase\intl\gdsintl.dll

Win64

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\win64_togo\ibtogo64.dll
- ☒ \$(IBREDISTDIR)\win64_togo\ibxml64.dll
- ☒ \$(IBREDISTDIR)\win64_togo\ib_util64.dll
- ☒ \$(IBREDISTDIR)\win64_togo\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\win64_togo\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\win64_togo\interbase\ADMIN.IB
- ☒ \$(IBREDISTDIR)\win64_togo\interbase\license.txt
- ☒ \$(IBREDISTDIR)\win64_togo\interbase\intl\gdsintl.dll

OSX64

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\macos64_togo\libibtogo.dylib
- ☒ \$(IBREDISTDIR)\macos64_togo\libib_util.dylib
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\ibconfig
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\admin.ib
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\license.txt
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\macos64_togo\interbase\intl\gdsintl

iOSDevice64

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\ios64_device\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\ios64_device\interbase\admin.ib
- ☒ \$(IBREDISTDIR)\ios64_device\interbase\license.txt
- ☒ \$(IBREDISTDIR)\ios64_device\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\ios64_device\interbase\ibconfig

Linux64

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\linux64_togo\libibtogo.so
- ☒ \$(IBREDISTDIR)\linux64_togo\libib_util.so
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\ibconfig
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\admin.ib
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\license.txt
- ☒ \$(IBREDISTDIR)\linux64_togo\interbase\intl\gdsintl

Android64

- ☒ \$(IBREDISTDIR)\reg_ibtogo.txt
- ☐ \$(IBREDISTDIR)\reg_iblite.txt
- ☒ \$(IBREDISTDIR)\android64\interbase\interbase.msg
- ☒ \$(IBREDISTDIR)\android64\interbase\admin.ib
- ☒ \$(IBREDISTDIR)\android64\interbase\license.txt
- ☒ \$(IBREDISTDIR)\android64\interbase\oss_license_notice.txt
- ☒ \$(IBREDISTDIR)\android64\interbase\ibconfig

Database file + platform files





Maintaining InterBase





Database maintenance

- Sweeping a database
- Shutting down the database to provide exclusive access to it
- Validating table fragments
- Preparing a corrupt database for backup
- Resolving transactions “in limbo” from a two-phase commit
- Validating and repairing the database structure





Backup /Restore

Database backup and restoration can also be used for:

- Erasing obsolete versions of database records
- Changing the database page size
- Changing the database from single-file to multifile
- Transferring a database from one operating system to another
- Backing up only a database's metadata to recreate an empty database

Increase your backup performance:

- Disable garbage collection if you are just going to replace the database immediately anyway; this can make the backup execute faster.
- Back up to a different disk drive.

Speed up restore performance:

- Restore from a different disk drive.
- Disable indexes on restore; this makes the restore execute faster, so you have a usable database quickly. You must manually activate the indexes after the restore is complete.



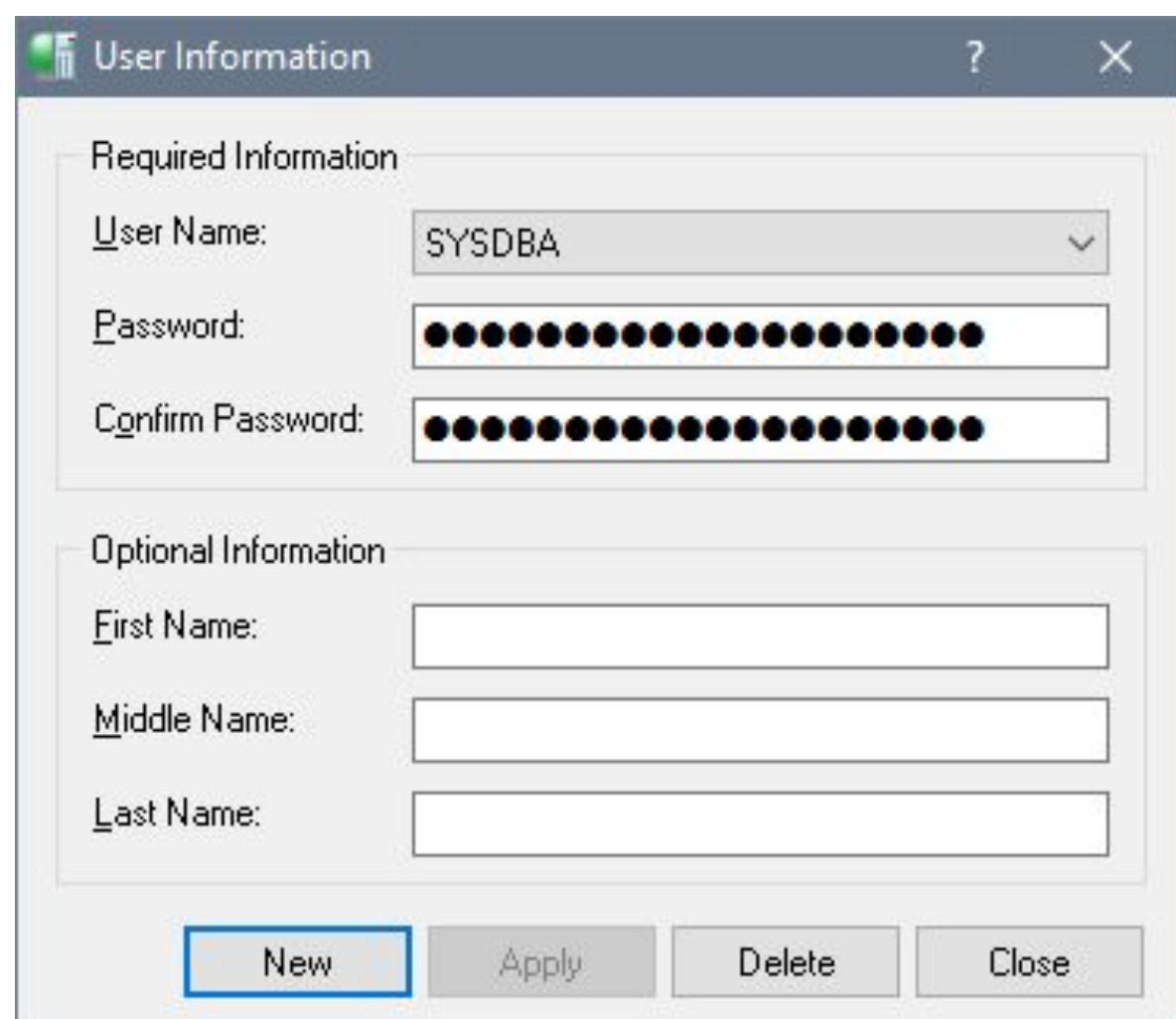


Managing Server Security

The InterBase server stores the names and passwords of its authorized users in a special security database that resides in the InterBase home directory. By default, it is named **admin.ib**.

Use IBConsole

or InterBase API



The image shows a 'User Information' dialog box with two sections: 'Required Information' and 'Optional Information'. In the 'Required Information' section, the 'User Name' dropdown is set to 'SYSDBA', and both the 'Password' and 'Confirm Password' fields are filled with black dots. The 'Optional Information' section contains three empty text boxes for 'First Name', 'Middle Name', and 'Last Name'. At the bottom, there are four buttons: 'New' (highlighted with a blue border), 'Apply', 'Delete', and 'Close'.

```
isc_add_user()  
isc_delete_user()  
isc_modify_user()
```





Journals and Journal Archiving

Journaling combines the protection of forced writes (also known as synchronous writes) with better performance, and also supports the improved disaster recovery provided by journal archiving.

How to determine the optimal journaling configuration of your system:

- The I/O speed of the device on which the journal files are created.
- The speed of concurrent creation of new journal files.
- Hardware requirements and ease of setup.

It is not necessary for journal archive storage files to be on a machine with InterBase installed.

```

CREATE JOURNAL [<journal-file-specification>] [LENGTH <number-of-pages> [PAGES]]
[CHECKPOINT LENGTH <number-of-pages> [PAGES]]
[CHECKPOINT INTERVAL <number-of-seconds> [SECONDS]]
[PAGE SIZE <number-of-bytes> [BYTES]]
[PAGE CACHE <number-of-buffers> [BUFFERS]]
[[NO] TIMESTAMP NAME]
[[NO] PREALLOCATE <number-of-pages> [PAGES]];

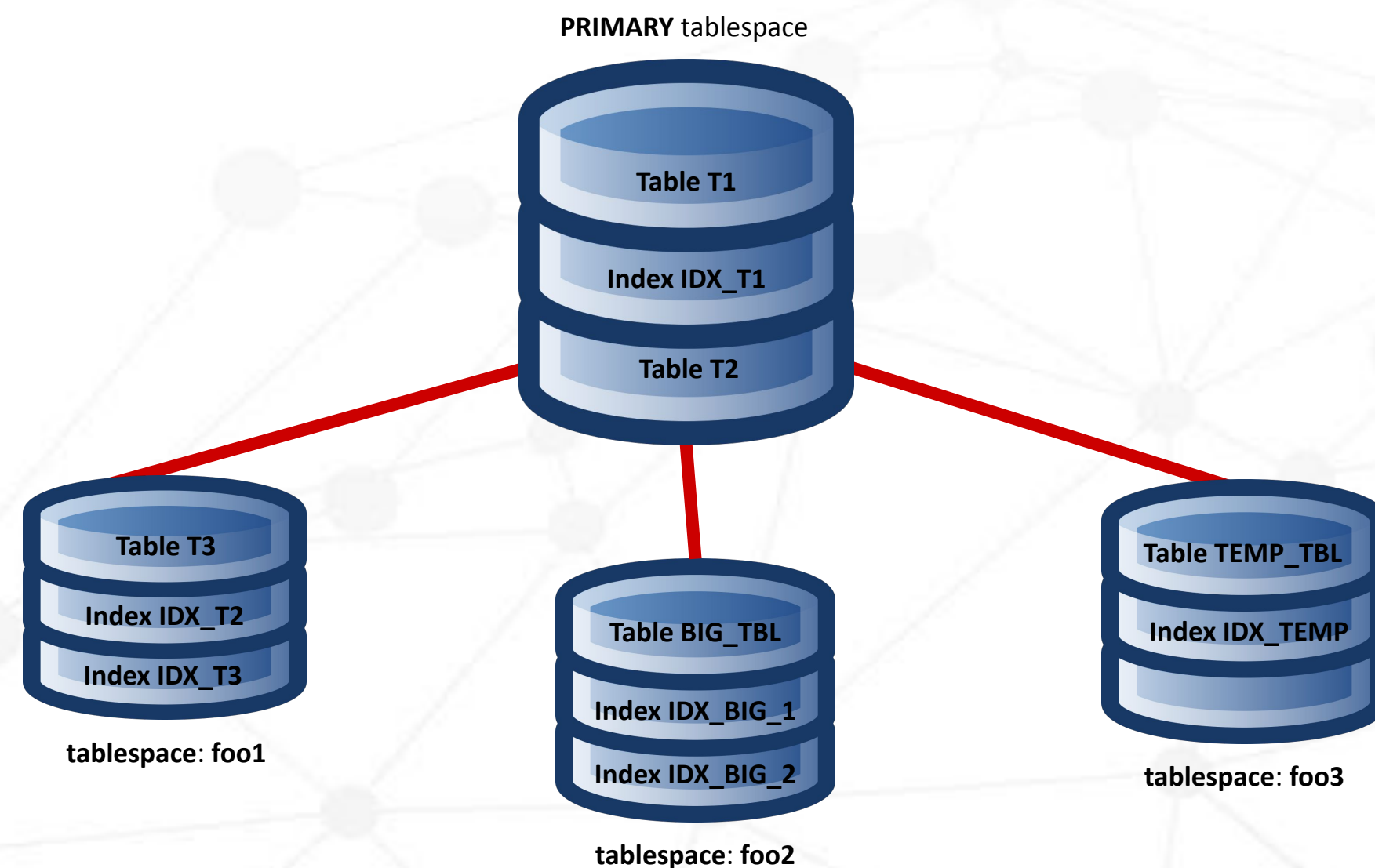
```





Tablespaces

- Tables and Indices can be located in individual tablespaces mapped to different physical drives or file systems.
- Better performance, balanced and optimized for use on your hardware.
- Lets you mix up your storage with both SSDs and HDDs
- *New* in InterBase 2020





Q&A

www.interbase.com

@InterBase

Mary.Kelly@embarcadero.com

