

InterBase Overview

InterBase® is the open source relational database that combines ease of use, low maintenance costs, and enterprise power. Since 1985 InterBase has provided the strength of a powerful, high-performance, proven architecture with the sophisticated technology applications need to be successful. Today, InterBase leads the industry again, offering free development and distribution rights and moving database use to the center of all application development.

Versioning Architecture for ultimate concurrency—readers never block writers.

Active database, including the most full featured trigger and stored procedure implementation.

Event Alerters—React to database changes without polling.

Exceptional ANSI SQL-92 compliance and full UNICODE support.

Rich data types—Blobs, multi-dimensional arrays.

InterClient—all-Java JDBC driver for low maintenance.

Designed for business critical distributed database environments, InterBase provides power and flexibility for Internet, mobile, and embedded database applications.

Scalable from Windows 95/98, Linux, HP/UX, Solaris, and other UNIX systems.

In January, Inprise announced that the new version of the InterBase database—InterBase 6.0—would be open source. This is your chance to read about what InterBase is, and what it does. You can then download the latest version of the software and then use and test the 6.0 release to see if it meets your requirements.

Business-critical tasks like stock trading, pharmaceuticals, network management, and aerospace demand performance. InterBase delivers their performance and remains practical and easy to deploy for small applications. Developers who want a sophisticated database with a small footprint, low maintenance cost, and high reliability choose InterBase. They depend on InterBase to manage their customers' data without the services of a highly trained and compensated DBA.

InterBase offers the best features of the database world—triggers, stored procedures, Blobs, event alerters, user defined functions, multi-dimensional arrays, two-phase commit, referential integrity, constraints, and a flexible set of transaction options. These features reduce development time and improve reliability.

Sophisticated Architecture Delivers Performance

The InterBase server implements a Multi-Generational Architecture [MGA]. This MGA provides unique 'versioning' capabilities that result in high data availability for transaction-processing users and decision-support users simultaneously.

Traditional database servers support the On-Line Transaction Processing (OLTP) model of database interaction, with many short, simple transactions. The InterBase MGA engine performs well on short OLTP-style transactions, but it excels in real world applications, outperforming other databases because concurrent long-duration, decision-support transactions do not degrade its performance.

"InterBase . . . outperformed all other SQL databases that we tested; nothing else came close. Its speed and small footprint, combined with its ability to handle data-intensive client/server applications, made it an easy choice for us."

—Russ LeMaster, Dover Elevator Systems

The versioning engine eliminates the need for transactions to lock the records they read, making them contention-free—**readers never block writers**. Unlike other databases, InterBase provides a time-consistent, repeatable result for every query, without special programming. Because long and short duration transactions can coexist, the InterBase versioning engine maximizes throughput for all transactions.

Multi-threaded Architecture

The InterBase server adds a multi-threaded architecture to the MGA, improving performance and optimizing the use of system resources, especially for large numbers of users. A shared server supports more clients on the same hardware and still improves the system responsiveness.

The multi-threaded architecture provides a shared data cache, reducing the amount of disk I/O for each application request. The server's shared metadata cache reduces the compilation costs for requests and makes procedures and triggers more efficient. User and database statistics kept by the server are useful for diagnosing application hot spots.

Java Enablement

Java® and InterBase are a natural pair. Features that make Java intriguing—simplicity, robustness, portability, and flexibility—are also characteristic of InterBase. Java applications access InterBase through InterClient, a high-performance JDBC driver.

InterClient is an all-Java driver, which can be an applet—nothing is installed on the client. Deployment is simple, since configuring machines with client libraries is unnecessary. InterClient makes upgrading easier as well, because database upgrades on the server never make your client obsolete.

Easy to Manage and Maintain

"The database we picked had to perform searches on millions of records. It had to scale to meet our projected requirements well into the future. Since our staffing levels are such that no one person can act as a full-time database administrator, we also needed an application that was largely self-maintaining. Finally, as a city agency, we were under strict budget and management constraints. Taking all of this into consideration, and after looking at every major database on the market, there was only one clear choice: InterBase."

—Al Porco, Division of Disease Intervention, City of New York

Most SQL database server products require expensive MIS staffs to install, tune, and manage them. The InterBase design doesn't require hours of maintenance or a PhD in InterBase tuning. When your applications must run without constant supervision or when your desktop database runs out of steam, InterBase is the clear choice.

Installs in Minutes

InterBase is simple to install. It has a small footprint, so you do not need a lot of free disk space. InterBase is self-tuning, so you won't be required to set hundreds of incomprehensible parameters. InterBase optimizes itself for you.

"As powerful as [InterBase] is, it's not hard to learn, and it's easy to set up and install."

—Peter Miller, Hospitality Data Systems

Lower Life Cycle Costs

When estimating all the costs over a product life cycle, remember the following:

- InterBase requires less memory than most database systems, so you will buy less RAM for high performance.
- Because of the InterBase tight code with its small footprint, you need less disk space.
- You will be productive quickly because InterBase adheres to industry standards.

- If you upgrade to a new operating system such as UNIX, you can redeploy without rewriting any of your database objects by backing up and restoring your database.

The life cycle of a product made with InterBase—from **conception to end-of-life**—has minimal cost, no matter how your application changes.

High Reliability for All of Your Applications

InterBase pioneered the concept of an active database, building advanced automation technology into the server's kernel. InterBase active database features include our patented event alerters, stored procedures, triggers, User-Defined Functions, and Binary Large Object (Blob™) filters. Together they move data processing steps to the server—where they are fastest and most reliable. Complementing this strong support for built-in business rules, InterBase ensures data reliability with declarative referential integrity, including cascading operations.

Event Alerters Automate Your Applications

Event alerters notify "interested parties" when specific changes occur in the database. An application registers interest in an event, then waits without polling the database until it is notified that the event has occurred. By eliminating polling, event alerters save system resources and make applications scale better.

Real-World Example

Problem: Stock tracking database needs constant polling to make sure inventory does not drop too low before more is purchased.

Solution: With InterBase, event alerters can be set up to register specific events, and notify the parties who have expressed their interest in these events when they occur. So when stock on widgets drops below 500, an e-mail message is sent to the purchasing manager letting her know to buy more of that item.

Triggers: Reusable Business Objects

Triggers store and enforce a company's business rules on the server. The server itself guarantees that every application using corporate data adheres to these rules. InterBase triggers automated responses to events on the server, and validates input data whenever a row is inserted, updated, or deleted from a table.

"InterBase . . . has the best implementation of modular, optionally-ordered, pre- and post-operation triggers."

—DBMS, July 1996

Stored Procedures: Reusable Business Processes

Stored procedures off-load common business tasks from the client to the server, causing major performance gains. Any InterBase application can use stored procedures. They encourage modular design, and make reuse and maintenance easier.

User-Defined Functions: Reusable Custom Features

User-Defined Functions (UDFs) give developers a means of extending the analytical capabilities of InterBase. They are reusable code, accessed from the server, and ensure data reliability and integrity. UDFs can process data themselves or call external services. InterBase provides a **default library** of User-Defined Functions that offers commonly used functions. This library includes:

- Math and trigonometry functions, including cos, sin, base, log, and more.
- String functions: difference, insert, substring, and more.

Declarative Referential Integrity Constraints

Declarative referential integrity constraints maintain relationships between records in your database efficiently and reliably. InterBase supports four categories of constraints:

- **Unique and primary key:** No two rows in a table have the same value for the set of key columns.
- **Referential Integrity:** Parent-child relationships between tables are synchronized. The declaration can include cascading updates and deletes.
- **Check:** The associated condition will be valid for every row in the table.
- **Domain:** Create a new subtype and specify a range of acceptable values, enumerate a list of values, provide default values, and set a data type. Any column declaration may reference a defined domain as an alias for a more sophisticated data type.

Powerful Data Types Increase Flexibility

Unstructured data is an essential element for an increasing number of applications. InterBase meets this need with multidimensional arrays and Blobs. They make InterBase the best choice for multimedia and scientific applications.

Binary Large Objects (Blobs)

In 1986 InterBase set the industry standard by storing sound, image, graphic, and binary information directly in the database using its Blob data types. Internet and telephony applications use Blobs and Blob filters to store and manage multimedia data. Blob filters are custom routines that transform the contents of a Blob from one state to another. Filters are ideal for compression and translation and add nothing to the processing load on the client.

Multidimensional Arrays

InterBase provides direct support for the multidimensional arrays used in scientific and financial applications. A single field in the database can hold an array of sixteen dimensions. For data that is inherently organized as arrays, InterBase simplifies database design and increases performance.

Distributed Database: Application Flexibility

When you need to move your desktop database to something more sophisticated or enlarge a small workgroup application so that several departments can use it, InterBase is ideal. It was designed for distributed database environments.

Multi-Database Access

InterBase is a truly distributed SQL database server that lets each database system query and return information to any other InterBase server.

Automatic Two-Phase Commit

Multi-database transactions require more than just the ability to connect to two databases. To be transactions, they must be consistent and atomic. InterBase includes a two-phase commit that ensures that distributed updates are consistent, automatically.

Distributed Two-Phase Commit Recovery

InterBase goes beyond a simple two-phase commit. It was the first database to provide distributed recovery from a failure during a two-phase commit. This ensures full recovery with no single point of failure, since the coordination of the commit is distributed among all the servers. A transaction that cannot commit across all servers is automatically rolled back on all servers.

Database Shadows

InterBase allows you to keep an exact duplicate of the original database, called a database shadow. This copy is maintained in real time by the InterBase server, and provides an immediately available backup in case of hardware failure. The shadow runs automatically, and adds a minimal performance penalty. Your control over the shadow includes its use of hard disk space and distribution across available devices.

ANSI SQL-92

Training developers is expensive and time consuming. Because InterBase delivers exceptional SQL-92 compatibility, it reduces the learning curve dramatically for new programmers moving to InterBase. Because InterBase uses SQL in all its features—stored procedures, triggers, constraints, and declarative Referential Integrity—you benefit from your developers' prior exposure to an industry-standard language. InterBase is SQL-92 entry-level compliant, with many intermediate level features and some SQL 3 features such as ROLES for group-level security.

International Character Set Support: UNICODE

InterBase supports UNICODE, the universal character code, and many international character sets for data storage and manipulation. Columns in the same table can be created with different character sets, allowing easy worldwide deployment. Languages supported by InterBase include Big 5 (Chinese), Korean, and all major European languages.

InterBase Can Grow With You

With InterBase, you not only have optimized performance across the most popular Windows, Linux, and UNIX platforms, but also across your company needs. InterBase crosses the spectrum from single-user databases to workgroups, to enterprise systems. As your business grows, InterBase grows with you.

InterBase Specifications

Integrity

- Declarative Primary Key
- Declarative Foreign Key
- Cascade Declarative Referential Integrity
- Domain and column-level Check constraints
- Trigger procedures with the following features:
 - Unlimited triggers per record change
 - Invoked before or after record insertion, deletion, or update
 - Multiple triggers per action, optionally ordered.
 - Forward-chaining (cascading triggers)

Concurrency Control

- Optimistic locking
- Data isolation levels: read consistency, read committed, and cursor stability
- Shared, and protected lock types for explicit table-level locking

Availability

- Online backups
- Immediate recovery after failure

Distributed Database

- Simultaneously connected databases - limited only by hardware
- Automatic distributed transaction processing via two-phase commit

Datatypes

- Character (fixed/variable length): up to 64Kb per field
- Integer (8, 16, and 32 bit)
- Floating point: single and double precision
- Date and time: January 1, 100 to December 31, 5491
- Date, Time, and Timestamp
- Fully Year 2000 compliant
- Multidimensional arrays; up to 16 dimensions per column
- Blob: unlimited size
- Import and export of ASCII fixed-length data
- Blob filters for compressing or translating Blob field data

Standards

- ANSI SQL-92 entry-level conformant
- ODBC rev. 2.0 (16-bit)
- ODBC rev. 3.0 (32-bit)
- UNICODE compliant

Database Capacity

- Maximum number of rows per table: approximately 2 billion

- Maximum size of a table: limited by system resources
- Maximum number of databases per system: limited by system resources
- Maximum number of active users possible per system: limited by system resources
- Maximum number of tables per database: 64K
- Maximum row size (excluding Blob): 64Kb

System Requirements

- Minimum RAM and disk space varies with operating system platform
- Networking hardware and software dependent on operating system platform.

New in InterBase 6.0

SQL-Delimited Identifiers

Delimited identifiers are database object names that are delimited by double quotes, eliminating restrictions on database object names.

InterBase Express

Borland Delphi and C++Builder developers can now use the InterBase Express (IBX) components to build applications with InterBase through the InterBase API, improving performance and capabilities.

Large Exact Numbers

Provides conformance with the SQL92 Standard for Numeric and Decimal datatypes with 10 to 18 digits of precision as a native 64-bit integer.

SQL Date, Time, and Timestamp

The standard DATE, TIME and TIME-STAMP datatypes replace the old DATE data type. The new types include new operators, such as CURRENT_TIMESTAMP, CURRENT_DATE, CURRENT_TIME, and EXTRACT(), allowing the return of specific date and time values.

Read-Only Database

InterBase databases can now be created as read only, either for security purposes, or for storage and access via read only media such as a CD-ROM.

Services API

A new API that allows developers to write applications that access and monitor the database server and its management functions.

Open Source

InterBase Free of Charge

Starting with the release of Version 6.0, InterBase Software Corporation will no longer charge a license, usage, or copying fee for InterBase.

InterBase Free to Use

The Free Software Foundation says “think of ‘free speech,’ not ‘free beer.’” In addition to the binary versions, the full source of InterBase will be online and available for download. You may read it to understand the product, use it for debugging, or modify it to suit your needs.

InterBase Public License

InterBase has adopted a clone of the Mozilla Public License V1.1. The primary requirement of the license is that if you change the code and offer it to others, you must also offer the source of the changes, for free or for a minimal charge.

The InterBase Community

The source of InterBase, its interfaces, and the utilities will be available through a CVS tree. You are invited to assist in the ongoing development of InterBase by contributing fixes and extensions to any of those products.