GBASE®

# GBase 8s Secure Database Management System
# Technical White Paper V8.8



www.gbase.cn

**GBase 8s Secure Database Management System Technical Whitepaper,**

**General Data Technology Co., Ltd.**



**Copyright Notice**



**Disclaimer**



**Communication Method**

General Data Technology Co., Ltd.

East Tower of Putian Innovation Industrial Park, No.22 Kaihua Road, Huayuan Industrial Park, Tianjin High-tech Zone, Tianjin, China

Tel: 400-013-9696 Email: info@gbase.cn

**Trademark Statement:**

# Contents

**1 Product Overview ..... 1**
1.1 Product Description ..... 1
1.2 Feature Overview ..... 1
1.3 Function Overview ..... 3
1.4 Introduction to Shared Storage Cluster ..... 5
**2 Technical Specifications ..... 6**
2.1 Support Platform ..... 6
2.2 Functional Indicators ..... 6
2.3 Performance Indicators ..... 7
**3 Product Architecture & Deployment Methods ..... 9**
3.1 Product Architecture ..... 9
3.1.1 Product Architecture ..... 9
3.1.2 Security Architecture ..... 13
3.1.3 Cluster Architecture ..... 16
3.2 Deployment Methods ..... 17
3.2.1 Standalone ..... 18
3.2.2 Shared Storage Cluster ..... 19
3.2.3 Disaster Recovery Cluster ..... 19
3.2.4 Multi-Site Deployment ..... 20
**4 Basic Features ..... 20**
4.1 Data Types ..... 20
4.2 Locks and Isolation Levels ..... 22
4.2.1 Locking Techniques ..... 22
4.2.2 Isolation Levels ..... 22
4.3 Sharding and Indexing ..... 23
4.3.1 Sharding Techniques ..... 23
4.3.2 Indexing Techniques ..... 23
4.4 Unstructured Data Support ..... 24
4.5 Backup and Recovery ..... 24
4.6 SQL Compatibility ..... 25
**5 Security Features ..... 26**
5.1.1 Identity Authentication Feature ..... 27
5.1.2 Autonomous Access Control ..... 27

5.1.3 Security Labels and Mandatory Access Control ..... 28
5.1.4 Data Integrity ..... 30
5.1.5 Data Security ..... 31
5.1.6 Object Reuse ..... 33
5.1.7 Security Audit ..... 33
5.1.8 Separation of User Privileges ..... 34
**6 Shared Storage Cluster ..... 35**
6.1 Operation Principle ..... 36
6.2 SSC Technical Features ..... 36
**7 High Availability Cluster ..... 37**
7.1 HAC Disaster Recovery Cluster ..... 37
7.2 Flexible High Availability Cluster ..... 39
7.2.1 Active-Active High Availability within the Same City ..... 40
7.2.2 Multi-Site Deployment High Availability ..... 41
**8 Product Performance ..... 42**
8.1 Efficient File Space and Disk Management ..... 43
8.2 Configurable Multi-Worker Thread Processing Capability ..... 44
8.3 Efficient Transaction Concurrency Control and Management Mechanisms ..... 44
8.4 Configurable and Efficient Memory Management Strategies ..... 44
8.5 Efficient Query Optimization Strategies ..... 45
8.6 Efficient Storage Encryption Mechanism ..... 45
**9 Interfaces and Management Tools ..... 45**
9.1 Development Interfaces ..... 46
9.1.1 ODBC ..... 46
9.1.2 JDBC ..... 46
9.1.3 ADO.NET ..... 46
9.1.4 GCI ..... 47
9.2 Programming Languages ..... 47
9.3 Frameworks ..... 47
9.4 Management Tools ..... 47
9.4.1 Enterprise Manager ..... 47
9.4.2 Security Management Tool ..... 48
9.4.3 Audit Management Tool ..... 49
9.4.4 Backup and Recovery Tool ..... 50
**10 Migration Capability ..... 51**
**11 Technical Services ..... 52**

# 1 Product Overview

## 1.1 Product Description

GBase 8s is an enterprise-grade transactional database developed by General Data Technology Co., Ltd. It is independently researched and has achieved maturity and stability, with proprietary intellectual property. The product complies with B2 security standards, and adheres to international database specifications and development interfaces such as SQL92/99, ODBC, JDBC, and ADO.NET. It supports shared storage clusters and multi-site deployments.

GBase 8s is suitable for OLTP (Online Transaction Processing) application scenarios, including critical core business systems in the financial and telecommunications industries. It is also applicable to industries with high requirements for information security and the operational and managerial information systems of large enterprises. In over 80% of scenarios, it can replace mainstream international databases and shared storage clusters.

## 1.2 Feature Overview

GBase 8s is a mature and stable hundred-terabyte financial-grade secure database that supports shared storage clusters and multi-site deployments.

- **Highest security level:**

GBase 8s has passed relevant tests for Level 4 security protection, commercial encryption, and the standard of the Electronics Science and Technology Institute. It

has obtained qualifications in data confidentiality, access control, attack resistance, and user behavior tracking. It supports encryption throughout the communication and data access processes, including software and smart card encryption, as well as separation of three powers.

- **Financial-grade high availability:**

GBase 8s comes with a rich set of high availability software components. It supports solutions such as same-city master/slave disaster recovery clusters (Same-City HAC) and remote disaster recovery clusters (Remote HAC), enabling fault-switching in milliseconds with a high availability level of 99.999%. It supports multi-site deployments, with remote disaster recovery distances of over 1000 kilometers.

- **Shared storage clusters:**

GBase 8s supports shared storage clusters, addressing the data high availability concerns in critical user businesses. It supports automatic fault switching in milliseconds. The cluster can accommodate up to 128 nodes, and users can achieve near-linear improvement in read performance by adding cluster nodes.

- **Extensive SQL compatibility:**

GBase 8s continuously improves its compatibility with mainstream database syntax, supporting a wider range of SQL usage methods compatible with multiple mainstream database syntaxes. This maximizes the reuse of users' and developers' database experience and reduces the learning curve for product usage.

- **International standards:**

GBase 8s is compatible with SQL92/99 standards and supports standard development

interfaces such as JDBC, ODBC, and .NET. It supports various commonly used character sets such as GB18030-2000, UTF8, and BIG5.

- **User-friendly:**

GBase 8s provides a wealth of management tools and supports graphical installation, management, and development. It includes a web-based database monitoring tool for real-time monitoring of database status, resource consumption, and health checks.

- **Low-cost migration:**

GBase 8s offers multi-level migration support. With up to 150 built-in mainstream database-compatible syntaxes, GBase 8s meets the requirements for seamless application migration. It provides various migration tools and real-time synchronization tools, ensuring business accuracy and stability during application migration and trial runs.

- **Strong technical ecosystem:**

GBase 8s continuously tracks mainstream development frameworks, building a strong technical ecosystem. Supported development frameworks include Hibernate, MyBatis, Activiti, XORM, EF Core, SQLAlchemy, etc.

## 1.3 Function Overview

**Table.** GBase 8s Product Function Summary

| Function | Detailed Description |
|---|---|
| Data Types | Supports a wide range of data types, including character, numeric, date, binary, boolean, and large objects. |
| Standard Support | Compatible with SQL92/99 standards. Supports standard interfaces such as ODBC, OLEDB, JDBC, ADO.NET. |

| | |
|---|---|
| Transaction Management | Supports ACID properties of transactions. Supported isolation levels include Dirty Read, Committed Read, Cursor Stability, Repeatable Read, and Last Committed Read. |
| Lock Mechanism | Supports exclusive locks, shared locks, escalation locks, and multi-level lock mechanisms at the database, table, page, row, byte, byte range, and key levels. |
| Reliability | Supports various methods of data backup and failure recovery. Enables online backup to achieve uninterrupted operation of the product. |
| Large Data Volume | Supports complex data types and massive data volumes. Supports storage and management of data up to hundreds of terabytes to meet user application requirements. |
| Data Storage Management | Provides convenient and reliable tools for managing data files and log files. |
| Schema Object Management | Provides management tools for schema objects such as tables, indexes, views, data constraints, stored procedures, and triggers. |
| Security | Supports identity authentication, independent access control, data storage encryption, data communication encryption, and security auditing. |
| Availability | Supports high availability deployment methods such as master/slave, shared storage, and remote disaster recovery. Supports multi-site high availability deployment with distances of over 1000 kilometers. Supports automatic failover and read-write separation. |
| Peripheral Tools | Supports enterprise manager, monitoring tools, migration tools, real-time synchronization tools, etc. |
| Programming Languages | .NET, C, C++, C#, Java, PHP, Ruby, Cobol, Perl, Python, etc. |
| Development Frameworks | SqlAlchemy, XORM, Hibernate, Mybatis, Activity, EFCore |
| Compatibility | Compatible with mainstream database SQL syntax, as well as procedural languages such as stored procedures and functions. |

## 1.4 Introduction to Shared Storage Cluster

To address issues such as server power outages, network failures, process termination, and database downtime that may occur in information systems across various industries, GBase 8s has introduced a high availability cluster solution based on shared storage. This solution allows the construction of a highly available database system on low-cost servers, enabling flexible application deployment and automatic failover. GBase 8s Shared Storage Cluster (SSC) provides services through multiple nodes to ensure system high availability. Data is stored on shared storage, and all nodes share the same data. The communication and SQL protocols provided by the shared storage cluster are the same as those of a standalone system, ensuring seamless migration of application systems. GBase 8s SSC has been widely used in various industries, including telecommunications, finance, power, rail and transportation, demonstrating its maturity and stability.

**The SSC features the following:**

- **High Availability:**

  When a node fails, other nodes automatically take over the service requests. The failover of the primary node occurs in milliseconds, and the failure of a secondary node does not affect database services.

- **Scalability:**

  GBase 8s Shared Storage Cluster supports up to 128 nodes, and users can increase the number of nodes online according to their needs. Read performance scales nearly linearly.

- **Application Transparency:**

  When used in conjunction with the GBase 8s Connection Manager, failover

does not affect business operations.

- **Cost-effectiveness:**

The features of GBase 8s SSC are included in the GBase 8s Enterprise Edition and are not billed separately. It can be directly deployed on shared storage arrays without the need for a clustered file system.

# 2 Technical Specifications

## 2.1 Support Platform

| Category | Content |
|---|---|
| Mainstream Operating Systems | Linux, Aix, Solaris, HP UNIX etc. |
| Storage | Supports local storage, centralized storage, distributed virtual storage, etc. |

## 2.2 Functional Indicators

| Technical Specification | Description |
|---|---|
| Common Data Types | Supports **numeric types:** INTEGER, SERIAL, SERIAL8, BIGSERIAL, INT8, SMALLINT, BIGINT, BIGSERIAL, DECIMAL, NUMERIC, REAL, SMALLFLOAT, FLOAT, DOUBLE PRECISION, MONEY; **character types:** CHAR, CHARACTER VARYING, VARCHAR, LVARCHAR, NCHAR, NVARCHAR; date types: DATE, DATETIME, INTERVAL; **boolean type:** BOOLEAN. |
| Structured Query Language | Supports SQL 92/99 standards, including standard DDL, DML, DQL, and other database operations. |
| Database Objects | Provides common database object operations such as creating, modifying, and deleting databases, tables, indexes, views, sequences, stored procedures, triggers, system functions, and user-defined functions. Supports creating and deleting database users and assigning/retracting user |

| Technical Specification | Description |
|---|---|
| | permissions. |
| Backup and Recovery | Supports online full backups, incremental backups, and data recovery. |
| High-Density Transaction Processing | Utilizes a multi-threaded architecture to effectively utilize CPU resources, providing high-concurrency user connections and operations. |
| Data Replication | Supports data replication for high availability (HAC) using logical log-based synchronous replication. |
| Sharding | Supports sharding with round-robin and expression-based shard strategies. |
| Character Sets | GB18030-2000, UTF8, BIG5, etc. |
| Large Object Types | Supports BYTE, TEXT, BLOB, CLOB. |
| Other Types | JSON, BSON, XML. |

## 2.3 Performance Indicators

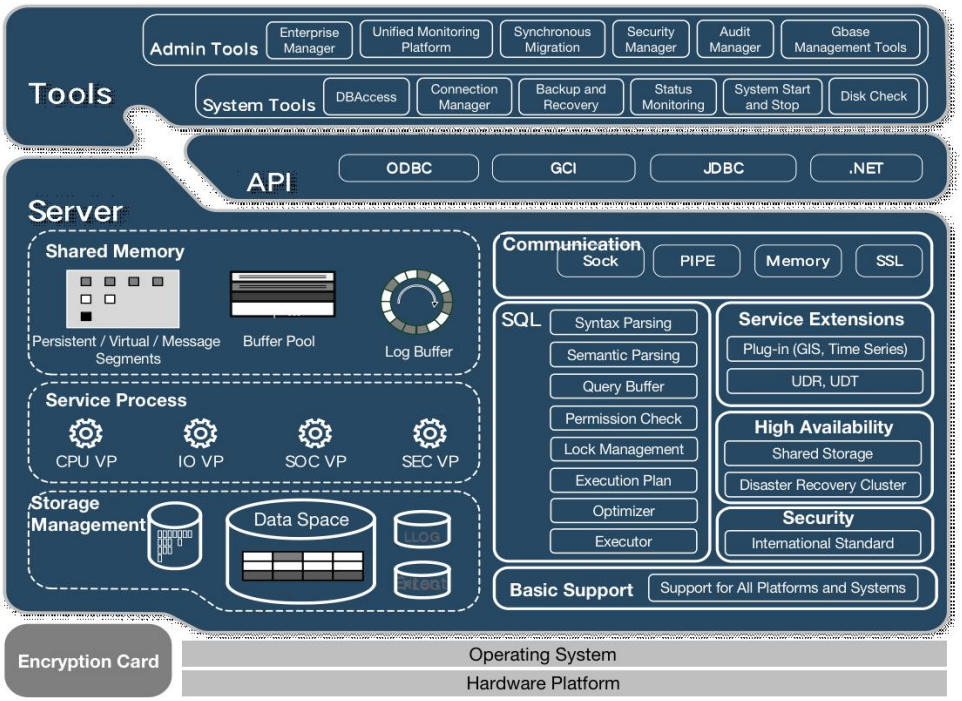
| Technical Specification | Description |
|---|---|
| Instances | 256 |
| Large Objects | 4TB |
| Data Compression Ratio | 1:5 |
| Numeric Precision | 32 |
| Number of Tables | Up to 65,536 per database |
| Internal Length of a Row in a Table | 32M bytes |
| Length of an Integer Type Column | 8 bytes |
| Number of Digits to Represent a Year in a Date Type Column | 4 digits |
| Number of Characters in a Username | 32 characters |
| Length of a CHAR Type Column | 32767 bytes |
| Length of a VARCHAR Type Column | 8000 bytes (default length, not the maximum limit) |
| Length of a Database Name | 128 characters |
| Length of a Table Name | 128 characters |
| Length of a Column Name | 128 characters |
| Length of an Index Name | 128 characters |
| Length of an Alias | 128 characters |
| Number of Data Rows per Page | 255 |
| Maximum Number of Rows in a Table or | 4,277,659,295 |

| Technical Specification | Description |
|---|---|
| Shard | |
| Maximum Number of Columns Supported | 32,767 |
| Maximum Length of a Single SQL Statement | 2GB |
| Maximum Number of Databases Supported by an Instance | 21,000,000 |
| Maximum Number of Tables Supported by an Instance | 477,102,080 |
| Maximum Number of Active Users Supported by an Instance | 32,767 |
| Maximum Number of Databases Accessible by a Single Session | Depends on the operating system: up to 32 for Linux/Unix and 8 for Windows |
| Maximum Number of Page Cleaning Processes | 128 |
| Maximum Number of Maximum Partitioned Data | 2K pagesize 1048445<br>4K pagesize 1048314 |
| Maximum Number of Tables Can Be Locked with Lock Table by a Single User | 32 |
| Maximum Space of a dbspace | 2K pagesize 4TB<br>4K pagesize 8TB |
| Supported Maximum Number of Chunks | 32,767 |
| Maximum Number of Data Pages Stored in a Single Chunk | 2,000,000,000 |
| Maximum Number of dbspaces and Storage Space Supported by an Instance | 2047/128PB |

# 3 Product Architecture & Deployment Methods

## 3.1 Product Architecture

### 3.1.1 Product Architecture




The system can be divided into three main components: Management Tools, Application Interfaces, and the Database Main Service.

- Management Tools: This includes graphical management tools and command-line management tools.
  - Graphical Management Tools:
    - Security Manager: Provides graphical interfaces for security-related tasks such as security labels and user authorization.

- Audit Manager: Allows opening and closing of security audits, setting audit labels, and viewing audit records.
- Unified Monitoring Platform: Displays the runtime status of the database system, allowing system maintainers and administrators to optimize, troubleshoot, and perform routine maintenance tasks.
- Migration Tool: Provides import tools for same-source or different-source data, such as migrating from Oracle to 8s.
- Enterprise Manager: A graphical management tool for database DBAs and regular users, providing graphical interfaces for managing database metadata, user data, and SQL queries/operations.
- GBase Management Tool: A graphical management tool for database DBAs and operators.

- Command-line Interfaces:
  - DBAccess: A command-line connection tool that allows executing DDL, DML, and DQL statements.
  - Connection Manager (CM): A distributed connection manager component for high availability.
  - Backup and Recovery Tool
  - Status Monitoring Tool
  - System Startup and Shutdown Tool
  - Disk Monitoring Tool

➢ Development Interfaces:

- JAVA Interface: JDBC.
- C Interface: ODBC, GCI (OCCI).
- ADO.NET.

➢ Database Service: The database service is a data management system

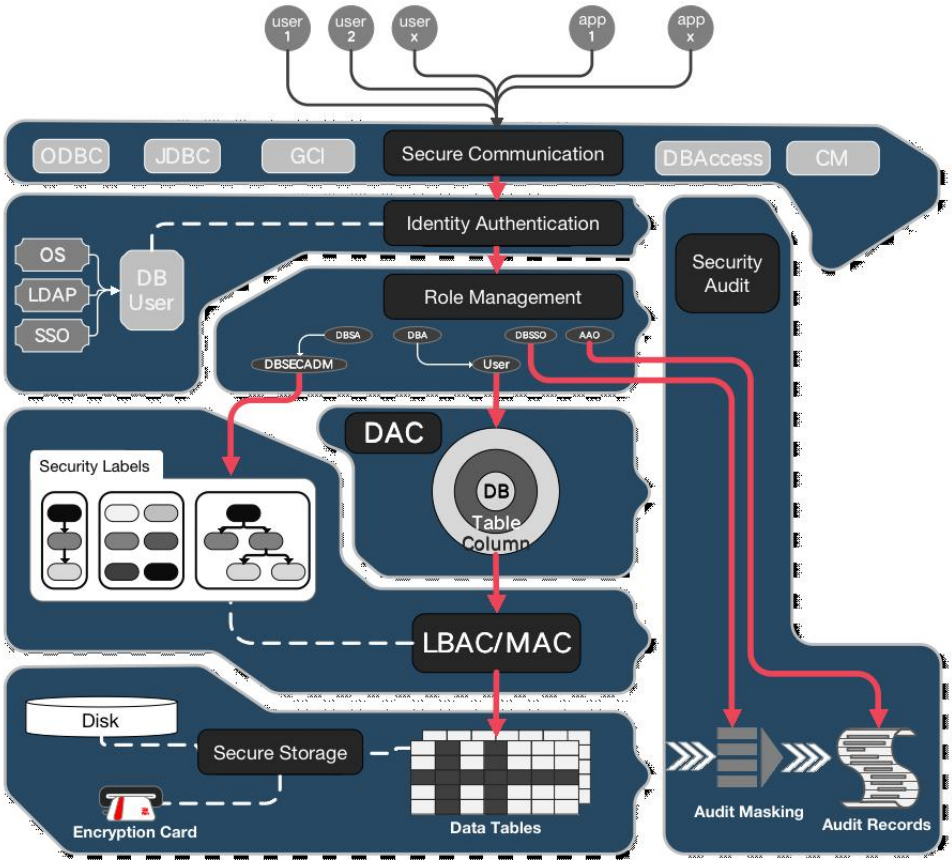
located between the user and the operating system. Its main functions include data definition, data manipulation, runtime management, and operations and maintenance. The architecture adopts a multi-process + shared memory design for efficient data management. The system consists of the following parts:

- GBase 8s utilizes shared memory mechanism for inter-process communication and data processing. Shared memory is also used as a data cache to reduce IO load and improve database concurrency.
- The service processes of GBase 8s are responsible for different functions and include user service processes, logical log service processes, physical log service processes, management service processes, security control service processes, IO service processes, audit service processes, and other related service processes. They handle tasks such as client security access, parsing, optimization, execution, and result retrieval. The resources of the main service processes can be dynamically adjusted based on the system load.
- Storage Management in GBase 8s provides database persistence capabilities and is an essential part of the database system. It includes functions such as physical storage medium management, file organization management, organization of records within files, data dictionary management, data index management, and large object storage, which manage the physical and logical elements of the database.
- Communication: Responsible for receiving SQL requests sent by business systems through application interfaces. After being processed and parsed by the SQL engine, this module returns the result set to the business system through the application interfaces.
- SQL Engine: The SQL engine module provides users with a high-level view of the data, including querying, inserting, deleting,

and modifying data. This module is also responsible for creating tables to store data and creating database objects (such as indexes, views, and stored procedures) for viewing, managing, and protecting data security.

- Basic Support Module: Includes encapsulation support for the underlying operating system, backend algorithms, and encapsulation of general-purpose function libraries, among other foundational functions.
- Security Module: Provides security protection features at the level of TCSES B2. It consists of components such as identity authentication, autonomous access control, labeling and mandatory access control, security auditing, trusted path, inference control, and self-protection of SSODB.
- High Availability Module: Provides high availability functionality for shared storage and disaster recovery clusters, meeting the requirements of business systems for high reliability.
- Service Extension Module: Provides management, storage, and usage of user-defined types and functions, and offers additional functionality through plugins, such as support for GIS information geospatial interfaces and time series.

## 3.1.2 Security Architecture




The security architecture of GBase 8s ensures that the database possesses security features. Users or applications access the database through secure communication using client-side authentication to confirm their identity and prevent unauthorized access. Role management confirms the roles of logged-in users, allowing them to perform tasks based on their role identity. The security administrator uses the LBAC module to define security labels and determine the data access policies for users. Data owners allocate access privileges to users for databases, tables, and fields using the DAC module. The system security officer defines audit policies to determine the content of auditing, while the audit operator enables, disables, configures auditing

functions, and analyzes audit records. Accessed data is stored on disk in encrypted form and is decrypted by the security storage module using hardware encryption cards for authorized user access.

**Secure Communication:** GBase 8s Secure Database uses the SSL security protocol, which protects the link layer using encryption algorithms and provides mutual authentication between the server and the client, ensuring the confidentiality and integrity of communication between them. Secure communication also protects the communication between the Connection Manager and clients in a high-availability environment, as well as the communication between servers.

**Identity Authentication:** GBase 8s supports conventional database user password authentication methods and also supports plug-in authentication modules such as Pluggable Authentication Module (PAM), Lightweight Directory Access Protocol (LDAP/GBase 8d), and Single Sign-On (SSO) for user authentication to the database.

**Role Management:** GBase 8s predefines default roles such as Database Security Administrator (DBSA), Database Security Administrator (DBSECADM), Database Single Sign-On (DBSSO), Database Audit Administrator (AAO), Database Administrator (DBA), and allows system administrators to create roles and assign users based on actual business requirements.

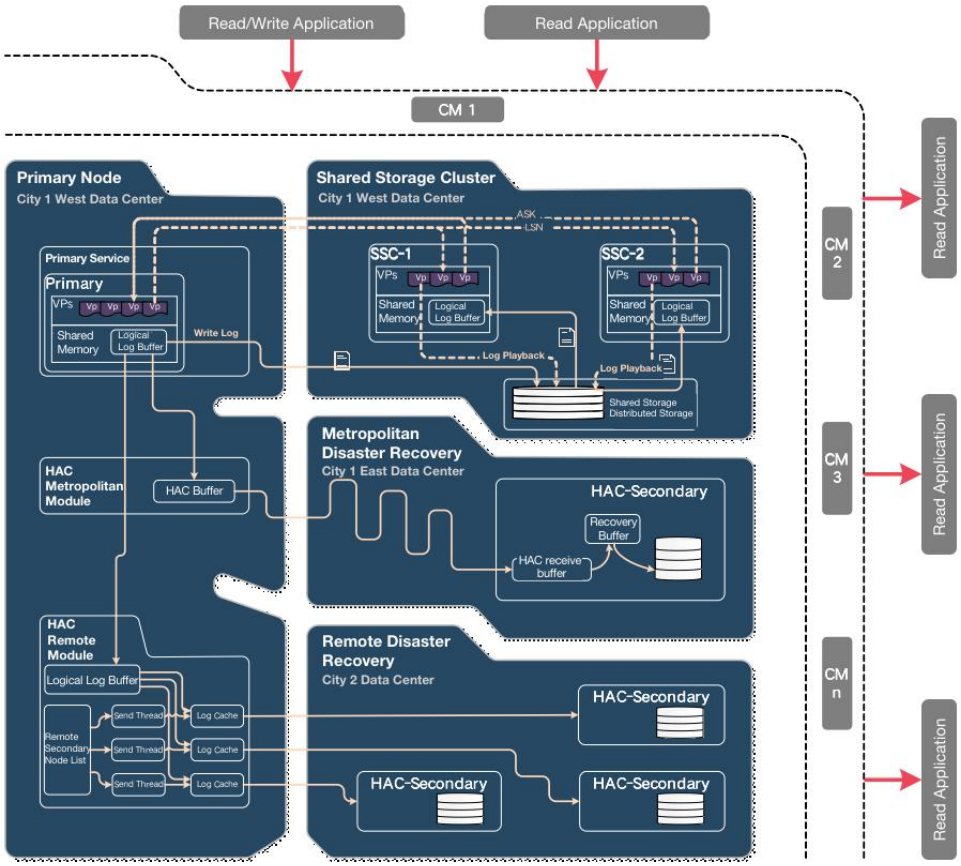
**Discretionary Access Control (DAC):** Discretionary access control is implemented at the database level, table level, and field level. Data owners and specified users can set different permission management for different granularity objects.

**Mandatory Access Control (LBAC/MAC):** GBase 8s uses a Label-Based Access Control system (LBAC) to achieve mandatory access control requirements. The security label consists of one or more ordered numerical components, unnecessary set components, and tree components representing hierarchical relationships. The level of mandatory access control reaches the row level.

**Security Storage:** Ensuring the confidentiality of stored data is one of the most important functions of a secure database. GBase 8s Secure Database encrypts data internally, performing encryption and decryption within the kernel storage engine of the database management system during physical I/O operations. This is known as transparent storage encryption since data is fully transparent to authorized users as encryption and decryption operations occur only during actual I/O operations.

**Security Auditing:** Audit operators set audit masks that can be configured separately for each user and globally, allowing the inclusion or exclusion of specific audit events. The audit mask can be set for a combination of 160 audit events. The audit operator manages the enabling, disabling, and configuration of auditing and analyzes audit records.

## 3.1.3 Cluster Architecture




GBase 8s system's high-availability cluster architecture consists of three types: Shared Storage Cluster, Disaster Recovery Cluster, and supports their combination to achieve a multi-site high-availability deployment. Cluster switching can be achieved through the Connection Manager (CM) component. Include:

- Disaster Recovery Cluster (HAC): The HAC technology replicates data from the primary server to a backup server. It is divided into two categories based on the distance of the backup: Local Disaster Recovery Cluster and Remote Disaster Recovery Cluster. The Remote Disaster Recovery Cluster is an extension of the local disaster recovery, providing a remote backup

solution. The Local Disaster Recovery Cluster supports synchronous, semi-synchronous, and asynchronous replication modes, while the Remote Disaster Recovery Cluster supports the asynchronous mode.

- Shared Storage Cluster (SSC): The SSC provides high availability by offering services through multiple nodes. Data is stored on shared storage, and all nodes share the same data. When the primary node fails, the backup node automatically and quickly takes over the services of the primary node.
- Connection Manager (CM): The CM is a lightweight connection management component situated between applications and the GBase 8s high-availability database cluster. It provides load balancing and failover capabilities based on the cluster's status. It supports two modes: proxy and redirection.

## 3.2 Deployment Methods

GBase 8s supports various deployment methods, including standalone, shared storage cluster, master-slave cluster, and multi-site cluster. It supports the deployment of applications and databases on the same server or different servers.

### 3.2.1 Standalone

Diagram of the application and database deployed on a single server:

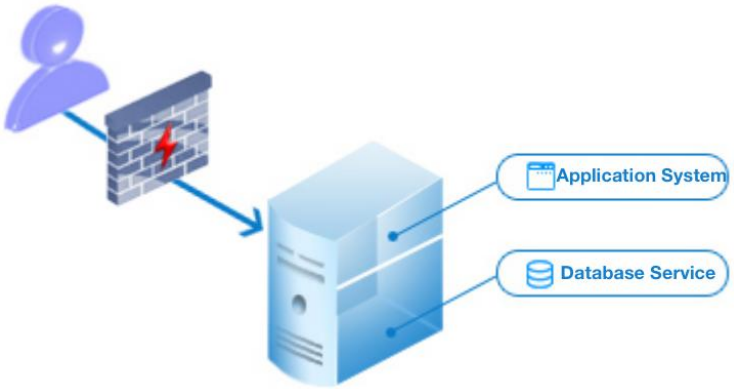
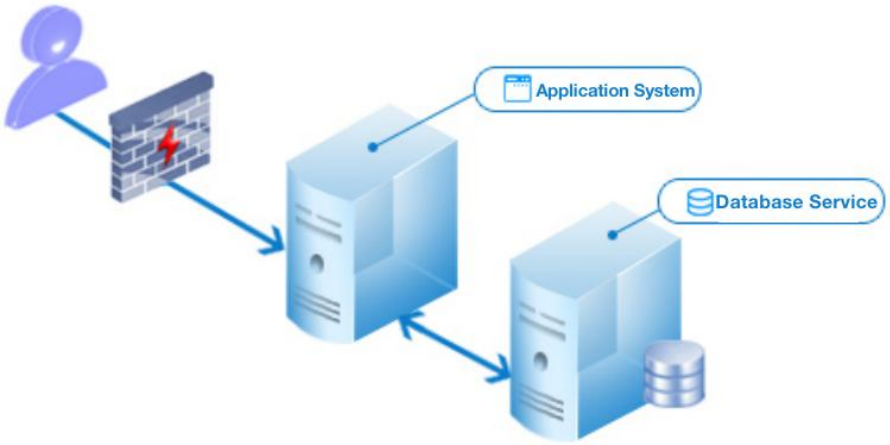
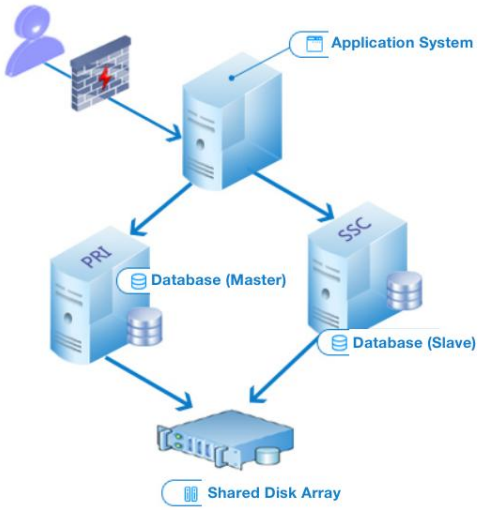



Diagram of the application and database deployed on different servers:



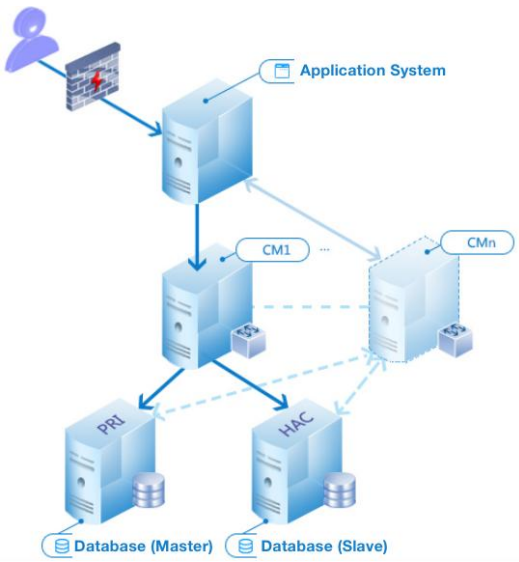
## 3.2.2 Shared Storage Cluster

Diagram of shared storage cluster deployment:



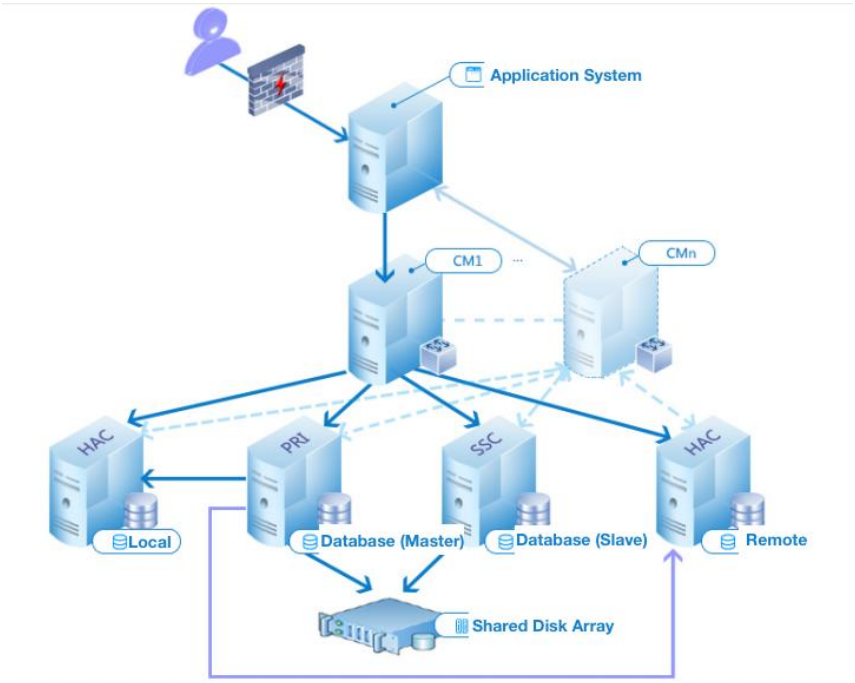

## 3.2.3 Disaster Recovery Cluster

Diagram of disaster recovery cluster deployment:



### 3.2.4 Multi-Site Deployment

Diagram of the deployment of a multi-site combined solution:




# 4 Basic Features

## 4.1 Data Types

GBase 8s has a comprehensive data type management system and includes all the data types commonly used in daily applications, fully satisfying the data management needs of enterprises. The supported built-in data types include:

- Numeric types: SMALLINT, INTEGER, INT8, BIGINT, DECIMAL, NUMERIC, FLOAT, SMALLFLOAT, REAL, DOUBLE, LONG, SERIAL, SERIAL8, BIGSERIAL, MONEY.

- Character types: CHAR, VARCHAR, LVARCHAR, NCHAR, NVARCHAR, CHARACTER VARYING.
- Date types: DATE, DATETIME, INTERVAL.
- Large object types: TEXT, BYTE, CLOB, BLOB.
- JSON/BSON data types.
- XML data type.
- Other types: BOOLEAN, ROW.

GBase 8s supports simple large objects and smart large objects as data types, which can store text, graphics, sound, and other content.

- Simple Large Object
  - ✧ Text: Stores large character objects, supporting a maximum size of 2GB.
  - ✧ Byte: Stores binary data, supporting a maximum size of 2GB.
- Smart Large Object
  - ✧ CLOB: Smart large character object, supporting a maximum size of 4TB.
  - ✧ BLOB: Smart binary object, supporting a maximum size of 4TB.

GBase 8s supports the definition of complex data types, mainly including the Row data type and collection data types.

- Row data type.
- Collection data types: Include Set, List, and Multiset.

GBase 8s provides user-defined data type (UDT) functionality, allowing users to define their own data types.

# 4.2 Locks and Isolation Levels

## 4.2.1 Locking Techniques

- Lock Granularity: GBase 8s provides six levels of lock granularity to meet the requirements of various concurrency scenarios.

| Lock Granularity | Description |
|---|---|
| Database Lock | Locks the entire database |
| Table Lock | Locks the entire table |
| Page Lock | Locks a whole page of data |
| Row Lock | Locks a single data row |
| Byte Lock | Locks a row containing VARCHAR data |
| Key Lock | Locks a key value in an index |

- Lock Types: GBase 8s supports multiple types of locks, including shared locks, exclusive locks, and promotion locks.
- Automatic Deadlock Resolution: GBase 8s intelligently manages deadlocks and provides automatic deadlock resolution functionality through relevant settings of database lock resources.

## 4.2.2 Isolation Levels

GBase 8s provides the following five isolation levels:

| Isolation Level | Description |
|---|---|
| Dirty Read | This concurrency level allows reading uncommitted data without locking any rows. |
| Committed Read | With this level, no rows are locked, but if someone performs an update or uses an exclusive lock on a row, it will fail. |
| Cursor Stability | Cursor Stability level adds shared locks on the selected rows, so other users cannot update them while someone is reading them. |

| | |
|---|---|
| Repeatable Read | For each read row, a shared lock is added, preventing them from being modified. Repeating the read operation will return the same records and values. |
| Last Committed Read | Similar to Committed Read, but when reading an already updated row, it reads the most recently committed record from the log. |

## 4.3 Sharding and Indexing

### 4.3.1 Sharding Techniques

GBase 8s supports two sharding methods: round-robin and expression-based sharding. Common sharding expressions include basic expressions, Mod operation expressions, Remainder, List, and Interval expressions. The Interval-based sharding strategy provided by GBase 8s automatically expands the shards based on the insertion records, offering a more flexible approach and reducing manual maintenance.

### 4.3.2 Indexing Techniques

GBase 8s supports various index structures, including B+ tree indexes and user-defined indexes. It offers multiple index types, such as unique indexes, function indexes, clustered indexes, and full-text indexes.

In addition to efficient query capabilities, GBase 8s indexes have the following features:

- Ability to create indexes online without impacting business operations.
- Support for Chinese and English word segmentation for full-text search. Incremental data automatically builds full-text indexes without manual

maintenance.

- Option to store indexes separately from tables.
- Index support for sharding, allowing for parallel scanning and improved scanning efficiency.

## 4.4 Unstructured Data Support

GBase 8s provides drivers for accessing unstructured data, offering excellent support for Web 2.0 websites with high performance, availability, and scalability.

Key features of GBase 8s in unstructured database support include:

- Support for storage and processing of unstructured data in JSON/BSON formats, with access through SQL, SQL functions, NoSQL database APIs, and NoSQL database command lines.
- Ability to establish connections and create indexes between unstructured tables.
- Transaction management and row-level locking capabilities for unstructured tables.

## 4.5 Backup and Recovery

GBase 8s backup is divided into three levels:

- Level 0 backup: Full backup.
- Level 1 backup: Incremental backup since the last Level 0 backup.
- Level 2 backup: Incremental backup since the last Level 1 backup.

Key features of GBase 8s backup and recovery functionality include:

- Flexible data import and export capabilities: Support for bulk import and export of entire databases, single tables, and multiple tables. Ability to import and export table structures (including all data objects) and table data separately. Support for exporting data in binary or text format for entire databases, single tables, or multiple tables, and the ability to import the exported data files into the database. Support for external tables to load and unload data.
- Backup and recovery support: Support for complete backup and recovery of single-node databases, incremental backup and recovery, and data space backup and recovery. Online backup and recovery are supported, with the ability to restore to a specific point in time. Backup and recovery of the database can be performed using storage management software (such as local backup and recovery software), with parallel execution capabilities.
- Support for database-level data file mirroring functionality.

## 4.6SQL Compatibility

GBase 8s supports a wider range of SQL usage methods that are compatible with multiple mainstream databases.

**Compatibility with mainstream databases:** Compatibility with data types such as varchar2 and timestamp. Compatibility with functions such as listagg() and sys_guid(). Compatibility with syntax such as with as, insert/first all. Support for adding comments to tables and setting default values for columns.

**Compatibility with procedural languages:** GBase 8s has achieved compatibility with commonly used syntax, allowing simple stored procedures to be created and

called in GBase 8s without modification. It supports syntax structures such as data type compatibility, variable declaration, assignment, sequential structure, selection structure, loop structure, static SQL, dynamic SQL, and exception handling.

**Compatibility with packages:** Support for package objects and a series of system-built packages starting with DBMS.

# 5 Security Features

GBase 8s provides and supports various security features, including identity authentication and verification, encrypted data storage, autonomous access control, security labeling, mandatory access control, data integrity protection, security auditing, and separation of powers. Please refer to the following diagram for an illustration of the security features provided and supported by GBase 8s.

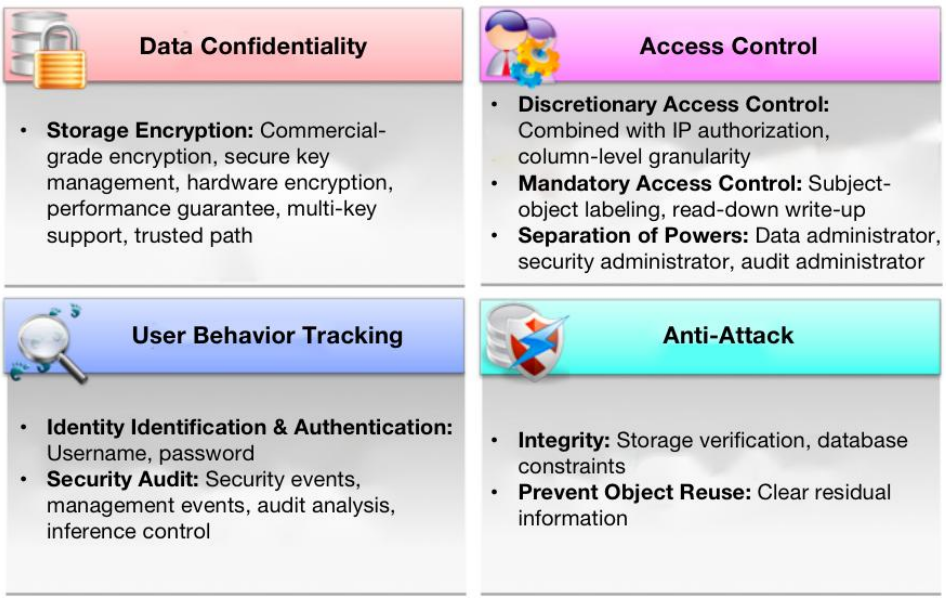



**Figure**. The security features provided and supported by GBase 8s

# 5.1.1 Identity Authentication Feature

## 5.1.1.1 User Identification

In GBase 8s, each database user has a unique and non-repeating user identification, ensuring the uniqueness of the user identification throughout the lifecycle of the DBMS.

## 5.1.1.2 User Authentication

User identity authentication is carried out according to the requirements of basic authentication, non-repudiation authentication, and one-time use authentication. Users must provide their user identification when using the DBMS and can only access it after passing the authentication process. User identity authentication adopts a dual authentication mechanism using user passwords and data certificates.

The passwords of database users are encrypted using a hash algorithm and stored in system tables in GBase 8s, ensuring the security of the passwords themselves.

# 5.1.2 Autonomous Access Control

Autonomous access control is implemented using an access control table, which follows the format shown in the table below:

Access Control Table

| | **Subject 1** | **Subject 2** | **……** | **Subject n** |
|---|---|---|---|---|
| **Object 1** | Operation 1.1 | Operation 1.2 | …… | Operation 1.n |
| **Object 2** | Operation 2.1 | Operation 2.2 | …… | Operation 2.n |
| **……** | …… | …… | …… | …… |
| **Object m** | Operation m.1 | Operation m.2 | …… | Operation m.n |

In the table, the subjects refer to users, and the objects include base tables, views, columns, stored procedures, functions, etc. The operations include SELECT, INSERT, UPDATE, DELETE, ALTER, INDEX, REFERENCE, EXECUTE, etc.

Based on the access control table, each subject is assigned certain operation permissions and can grant (or revoke) permissions to another subject, which is called authorization.

When a subject accesses an object, autonomous access control checks whether the operation is allowed based on the access control table. If the operation is allowed, the access is considered legitimate; otherwise, it is considered an unauthorized operation, and the access is denied.

Currently, the granularity of autonomous access control is at the user level for subjects and at the field (attribute) level for objects.

## 5.1.3 Security Labels and Mandatory Access Control

### 5.1.3.1 Security Labels

Both subjects (database users) and objects (data objects) in GBase 8s need to be labeled with sensitivity labels, referred to as labels. Labels consist of security level labels and category labels. Security level labels are represented by positive integers, while category labels are represented by sets.

The security administrator responsible for mandatory access control creates global levels and global categories and utilizes the created level and category labels for subjects (agents) and objects in the system.

### 5.1.3.2 Mandatory Access Control (MAC)

Mandatory Access Control (MAC) provides control over the sharing of objects (data objects) among subjects (database users). Unlike discretionary access control (DAC), which is managed by the object owners, MAC is managed by the security administrator. MAC prevents various direct and indirect attacks by enforcing access control restrictions that cannot be bypassed. A user is not authorized to grant access to any data resource to another user. GBase 8s adds security labels and mandatory access control features to the traditional discretionary access control foundation.

GBase 8s's mandatory access control model is based on the BLP (Bell-La Padula) model with appropriate extensions. The key elements of the model include:

- Subjects and Objects:

  In GBase 8s, subjects refer to database users, and objects include data tables, views, stored procedures, and functions.

- Security Labels (Hierarchical Levels and Non-Hierarchical Categories):

  Security labels, also known as labels, are assigned to each subject and object. Labels consist of security level labels and category labels. Security level labels are represented by positive integers, while category labels are represented by sets. A label is composed of a binary tuple of hierarchical level label (I) and non-hierarchical category label (S), which have a partial order relationship.

  For L1 = (I1, S1) and L2 = (I2, S2), L1 ≤ L2 holds if and only if (I1 ≤ I2) ∧ (S1 Í S2).

- Mandatory Access Control Security Policies:

  The basic security policy of the Bell-La Padula model is "no read up, no write

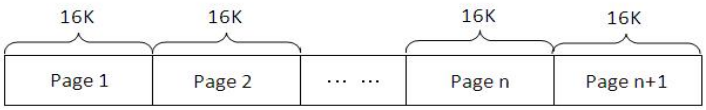
down." In GBase 8s, the security policy is expanded for practicality and flexibility. In addition to the read-down and write-up principles, there are also read-up and write-down principles. The write operation is further divided into insert and modify/delete operations, forming a security access policy for read-down and write, among others.

## 5.1.4 Data Integrity

GBase 8s provides a set of robust methods to ensure the integrity of the database.

- **Physical Storage Integrity Protection:**

Data files in GBase 8s are stored in a page-oriented manner, with each page being 16KB in size. The schematic representation of data file storage is as follows:




The storage structure of each page is as follows:


The Page Trailer stores the checksum information of the page. When data is written to the page, the checksum information is written to the Page Trailer. When data is read from the page, the checksum of the page is recalculated and compared with the checksum information in the Page Trailer. If they are not equal, it indicates that the

integrity of the page has been compromised. If they are equal, it indicates that the data integrity of the page is ensured and it can be accessed normally. GBase 8s checks the integrity of user data stored in the database in the form of page checksum information.

- **ACID Transaction Processing Model:**

GBase 8s supports ACID transaction properties to ensure the data integrity and validity of each independent transaction. It has a comprehensive commit and rollback mechanism. By combining redo logs and rollback segments, it ensures data consistency during disaster recovery.

- **Foreign Key Functionality:**

GBase 8s supports foreign keys to ensure referential integrity of the data.

- **Constraint Functionality:**

GBase 8s supports altering the constraint and unique index filtering mode, enabling or disabling constraints, indexes, and triggers, or bypassing referential integrity checks of foreign key constraints when resetting their constraint modes.

## 5.1.5 Data Security

### 5.1.5.1 Data Storage Security

GBase 8s utilizes in-database encryption for data storage, which is implemented at the kernel storage engine level of the database management system. This encryption is transparent to authorized users and is referred to as transparent encryption.

The existing storage encryption includes the following technical features:

- **Transparent Encryption:** Storage encryption is implemented by the database kernel in the backend secure storage engine, which is completely transparent to authorized users. It does not affect the frontend operations of legitimate database users and does not result in any loss of functionality.
- **Hardware Encryption:** It provides high-strength encryption functions and effective key management for secure databases through integrated cryptographic cards approved by the National Cryptography Administration.
  - PCI/PCI-E standard interface
  - Symmetric encryption algorithm SM1
  - Fast encryption and decryption operation speed (greater than 1000Mbps)
- Robust Key Protection: It adopts a multi-level key management mechanism, with the main key stored inside secure hardware. Under normal operation, the key does not appear in plaintext outside the encryption card.
- No Data Expansion in Encrypted Data: The ciphertext resulting from encryption has the same size as the plaintext, preventing any data expansion issues.

Due to reasonable optimization mechanisms, the overall performance degradation of the storage encryption feature in GBase 8s is less than 15%.

## 5.1.6 Object Reuse

Object reuse refers to the revocation of all authorizations of the information contained in an object in the free storage space of a trusted computing base of a computer information system before assigning or reallocating the object to another subject. When a subject gains access to a released object, it cannot obtain any information generated by the previous subject's activities. The object reuse feature prevents information leakage when important object media are reallocated to other subjects.

This functionality is automatically implemented by the GBase 8s system. After each resource allocation, the system automatically clears any residual information in the resource. These resources include memory units and disk areas, thereby avoiding security risks associated with object reuse.

**Memory Units:** Unnecessary information in memory will be cleared and deleted, ensuring that unwanted resource information does not remain in memory.

**Disk Areas:** Information remaining in disk areas will be automatically deleted when it is no longer needed. For example, temporary table files created on disk during a session will be automatically removed when the session is closed.

## 5.1.7 Security Audit

GBase 8s provides a security auditing mechanism to record user activities in the database. These activities can include data modifications, data displays, configuration of auditing and security policies, etc. By recording these activities, suspicious behavior can be analyzed.

For independent auditing, dedicated auditors are responsible for auditing management. Auditors can utilize a specialized auditing interface to select auditing events, review relevant auditing data, and handle alert signals.

### 5.1.8 Separation of User Privileges

Traditional commercial databases often define a super administrator with supreme authority, capable of performing any database function and managing any data. However, this creates security vulnerabilities due to the lack of constraints on privileged users. It is challenging to mitigate these security loopholes solely from an application perspective.

GBase 8s, the secure data management system, follows the principle of least privilege, granting users only the minimum necessary privileges required to fulfill their designated tasks. Based on this security policy, the database management system redefines its user roles, transforming the single super database administrator into three distinct roles: Security Administrator, Audit Administrator, and Data Administrator. Each role has specific responsibilities, and it is expected that their authority does not overlap with the other two, thereby achieving the principle of separation of powers within the entire database system.

The specific allocation of responsibilities for these three user roles is as follows: The Security Administrator primarily handles system security (marking) management, the Audit Administrator is responsible for system auditing functions, and the Data Administrator focuses on autonomous access control (DAC) and system maintenance management. These three types of administrator users have well-defined roles, mutually restraining and coordinating with each other, to collectively accomplish the

database's security management functions.

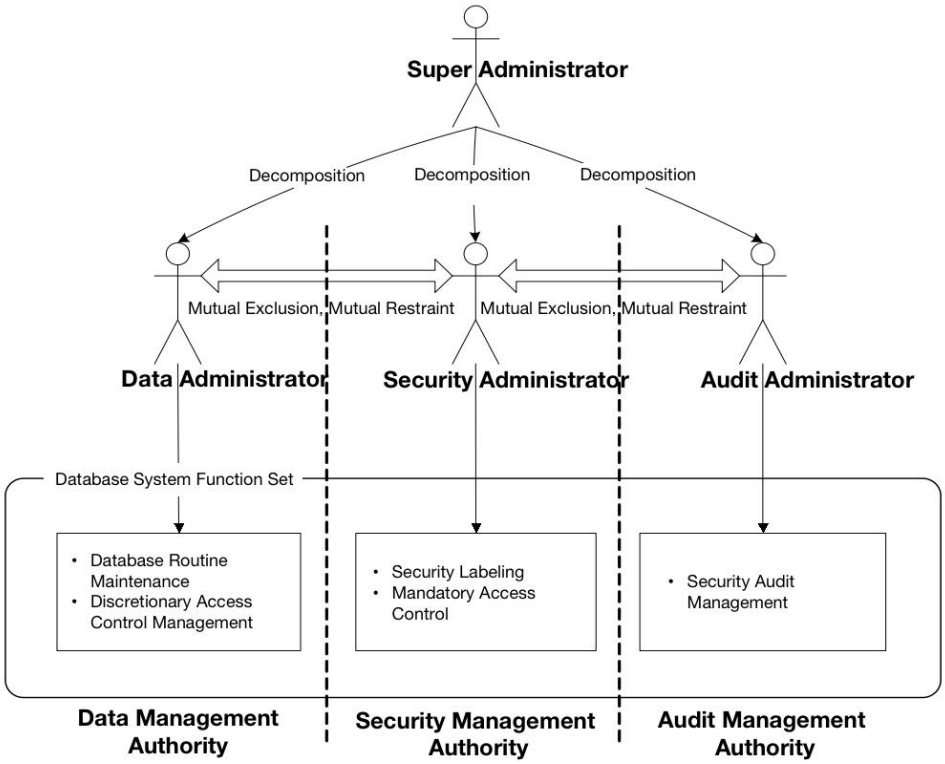


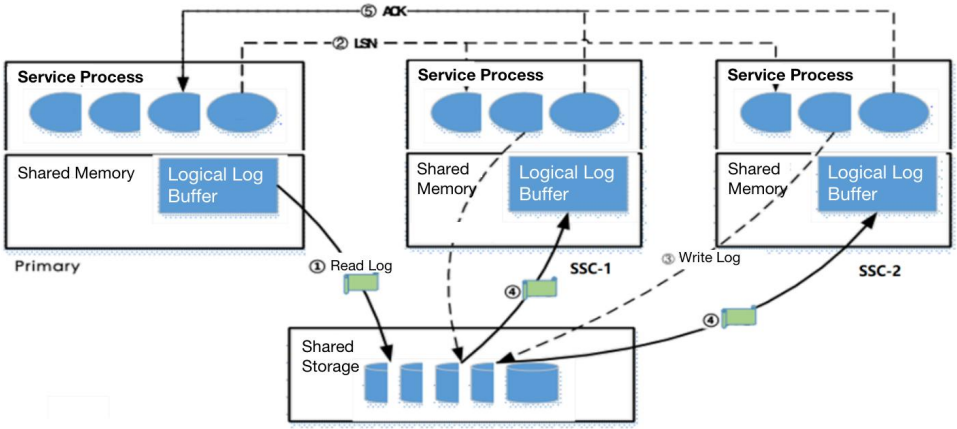

**Figure 4.2.** Illustrates the concept of the Separation of User Privileges in GBase 8s.

# 6 Shared Storage Cluster

The Shared Storage High Availability Cluster (SSC) utilizes shared disk technology to achieve high availability of compute nodes. Data is stored only once, effectively utilizing hardware resources and avoiding data duplication. Each node in the SSC cluster can provide services, making it suitable for scenarios with low write and high read demands that require linear scalability. The cluster can support a maximum of 128 nodes. SSC is characterized by its simplicity of installation, application transparency, automatic failover, and cost-effectiveness.

## 6.1 Operation Principle

The consistency of primary and standby nodes in SSC is achieved through LSN (Log Sequence Number) synchronization and logical log replay. The primary node receives business requests, writes logical logs, and sends the corresponding LSN numbers to the standby nodes. Upon receiving the LSN number from the primary node, the standby nodes read the corresponding logical logs from disk into the buffer for replay. This ensures data consistency between the primary and standby nodes. Changes to the data in the standby node's buffer are not persisted to disk.




## 6.2 SSC Technical Features

1) SSC offers scalability and can be easily expanded within a certain range according to business needs.

2) The relationship between nodes is symmetrical. When a node fails, other nodes in the cluster take over the failed node's work.

3) SSC deployment is simple, allowing for easy configuration of multiple SSC instances.

4) When used in conjunction with a connection manager, SSC enables load balancing and automatic failover.

5) Application transparency: Client applications can use SSC just like they would use a single database.

6) Cost-effective: The primary and standby nodes share the same storage, saving on storage costs.

# 7 High Availability Cluster

GBase 8s provides industry-leading high availability cluster (HAC) technology, including SSC and ER, to achieve automatic failover within seconds, resulting in a database availability of 99.999%. It has the following characteristics:

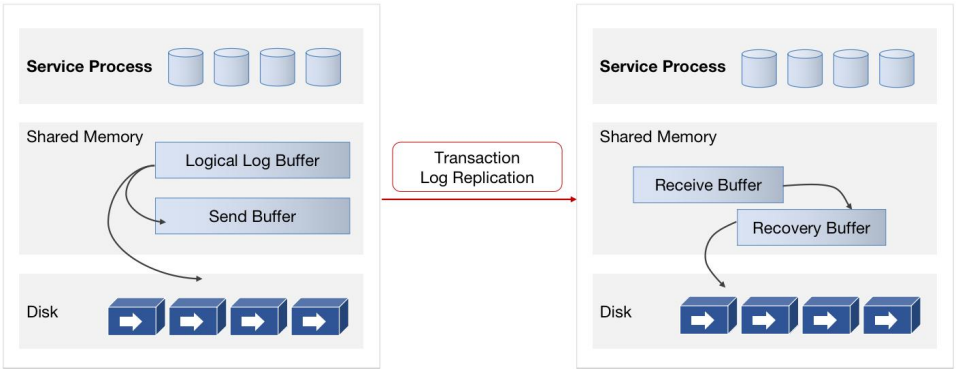
- By combining different schemes, GBase 8s can build multi-site and active-active high availability clusters within the same city.
- It supports long-distance disaster recovery functionality, spanning up to 3000 kilometers.
- It provides a connection manager (CM) component for failover and load balancing capabilities, transparent to applications.
- No need to purchase any third-party products, requires low bandwidth, and is easy to maintain.

## 7.1 HAC Disaster Recovery Cluster

HAC achieves node high availability using logical log replication technology. It can be divided into a local disaster recovery cluster and a remote disaster recovery cluster based on the distance. The local disaster recovery cluster supports a one-to-one

deployment of primary and standby nodes, while the remote disaster recovery cluster supports a one-to-many deployment. In the HAC cluster, the primary node handles read and write operations, while the standby node handles read operations such as queries and reports. HAC is characterized by its simplicity of installation, application transparency, automatic failover, and no additional costs.

HAC ensures the consistency of primary and standby node states through logical log replication. Therefore, it requires the primary and standby servers to have the same database version, and the hardware and operating system versions should ideally be the same. During operation, the HAC primary node receives business requests, writes logical logs, places the logs in a send buffer, and sends them to the standby nodes via a local area network or wide area network. The standby nodes receive the logical logs and replay them in memory, ensuring data consistency between the primary and standby nodes. The cluster switches between the primary and standby nodes through the CM (Connection Manager).




HAC is an architecture based on logical log technology, and the log update mode supports both synchronous and asynchronous modes.

- **Synchronous mode:** In synchronous mode, the transactions on the standby instance are always consistent with the primary instance, ensuring no data

loss in the event of a failure. It is suitable for local disaster recovery clusters where the primary and standby nodes are in close proximity (same data center or across data centers).

- **Semi-synchronous mode:** In semi-synchronous mode, the primary instance ensures that the logs have been replayed on the standby instance before flushing them to disk. It is suitable for local disaster recovery clusters where the primary and standby nodes are tens or hundreds of kilometers apart.
- **Asynchronous mode:** In asynchronous mode, the primary instance does not need to receive confirmation from the standby instance before flushing the logs to disk, allowing for faster log replication. It is suitable for remote disaster recovery clusters where the primary and standby nodes are thousands of kilometers apart.

Additionally, system administrators can configure and improve the synchronous mode of the HAC high availability cluster to balance system performance and data protection.

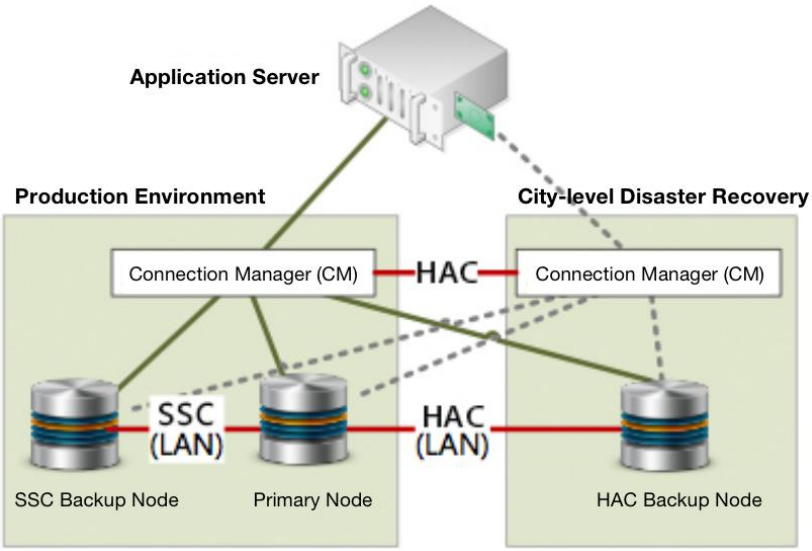
## 7.2 Flexible High Availability Cluster

GBase 8s' high availability solution supports combination usage. For example, using SSC+HAC for local disaster recovery technology to build an active-active high availability architecture within the same city, and using SSC+HAC for remote disaster recovery technology to build a two-site three-center high availability architecture.

| Applicable Scenarios | SSC | Local HAC | Remote HAC |
|---|---|---|---|
| **Network Connection** | Fiber | Internet or Dedicated Line | Fiber |
| **Distance Between Nodes** | Within meters, same data center | Within hundreds of kilometers, same city | Thousands of kilometers apart |
| **Bandwidth Requirement** | High | High | Low |
| **Maximum Number of Standby Nodes** | Multiple | 1 | Multiple |
| **Storage Devices** | Shared storage device | Independent storage device | Independent storage device |
| **Ability to Withstand Server Hardware/Software Failures** | Yes | Yes | Yes |
| **Ability to Withstand Natural Disasters** | No | Yes | Yes |

### 7.2.1 Active-Active High Availability within the Same City

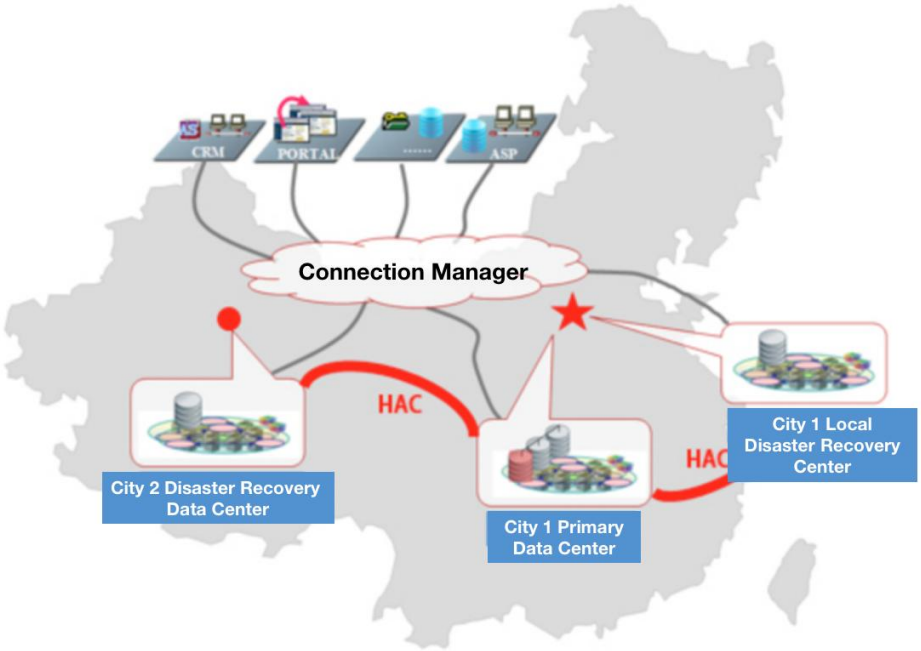
To meet the high availability requirements of enterprise core business systems, GBase 8s combines SSC cluster and local HAC cluster to create an active-active high availability solution within the same city. The overall architecture of this solution is shown in the diagram below:




In this solution, the SSC cluster is deployed in a one-master, multiple-standby node configuration, while the HAC cluster is deployed in a one-master, one-standby node configuration. The SSC cluster is deployed within the same data center, and the HAC cluster is deployed in another data center within a hundred kilometers. Failover is achieved through the connection manager. The standby nodes in the HAC cluster serve as read-only nodes, typically used for data querying, reporting, and relieving the workload of the primary system. The HAC configuration is primarily used to provide high availability and hot standby functionality.

## 7.2.2 Multi-Site Deployment High Availability

To meet the 99.999% high availability requirements of financial and telecommunications core businesses, GBase 8s utilizes its powerful clustering capabilities and combines SSC cluster with HAC cluster to create a multi-site deployment high availability solution. The overall architecture of this solution is illustrated in the diagram below:




In this solution, a 2-node SSC cluster is deployed within the same data center, consisting of one master and one standby node. The master node in the SSC cluster forms a one-master, one-standby local HAC cluster with the standby node deployed in another data center within the same city. The master node in the SSC cluster also forms a one-master, one-standby remote HAC cluster with the standby node deployed in a distant city. The SSC nodes are equivalent and support both read and write operations, while the standby nodes in the HAC clusters provide read-only functionality.

# 8 Product Performance

The goal of GBase 8s is to achieve a high-performance and secure database system that excels in transaction processing for online transaction processing (OLTP)

applications. To achieve high performance while ensuring system security, GBase 8s employs the following mechanisms:

- Efficient file space and disk management
- Configurable multi-worker thread processing capability
- Efficient transaction concurrency control and management mechanisms
- Configurable and efficient memory management strategies
- Efficient query optimization strategies
- Transparent storage encryption mechanism

## 8.1 Efficient File Space and Disk Management

GBase 8s builds and writes database files by establishing multiple disk I/O threads that simulate asynchronous disk I/O. The number of I/O threads is configurable and separate from the worker threads mentioned later. These threads are dedicated to disk I/O operations. Two mature heuristic prefetching techniques, namely sequential prefetching and random prefetching, are used to improve disk operation speed. Sequential prefetching means that if GBase 8s detects sequential access to tablespace segments, it proactively requests batch reading of database pages from the I/O system. Random prefetching refers to requesting read operations for the remaining space when GBase 8s determines that some space in the tablespace needs to be fully read into the buffer pool for processing.

GBase 8s can use a "raw disk" partition as a tablespace data file, improving performance by bypassing the file system and executing non-cached I/O.

## 8.2 Configurable Multi-Worker Thread Processing Capability

GBase 8s allows users to configure the number of worker threads that are shared by the entire system and are not specific to a particular database connection. If a database operation is blocked due to any reason (e.g., lock waiting), the corresponding worker thread will immediately be utilized to perform other database operations instead of being blocked. Additionally, for single-CPU and multi-CPU devices, the system efficiently shares physical resources such as memory and data among threads, achieving excellent concurrency processing capability.

## 8.3 Efficient Transaction Concurrency Control and Management Mechanisms

GBase 8s adopts a lock technique primarily based on row-level locks and supplemented by table-level locks to achieve efficient transaction processing. These techniques greatly enhance transaction concurrency while ensuring the ACID properties of transactions.

By implementing deadlock detection, GBase 8s automatically detects deadlocks and rolls back one or more transactions to prevent deadlocks. It also attempts to extract smaller transactions for rollback to reduce the significant disk flushing caused by rollbacks and improve performance.

## 8.4 Configurable and Efficient Memory Management Strategies

GBase 8s provides various caching mechanisms for different database operations, including data buffer caches, query buffer caches, key caches, and insert buffers. The

coordinated use of these buffers effectively utilizes limited system memory, greatly enhancing system performance. Furthermore, fine-tuning of system performance can be achieved by adjusting read buffers, sequential read buffers, sorting buffers, and other parameters.

## 8.5 Efficient Query Optimization Strategies

GBase 8s employs a cost-based query optimization strategy that focuses on efficient utilization of indexes. The query subsystem performs complex transformations on input query statements to generate different query plans. Then, based on the estimation of the execution cost for various query plans considering system resources such as I/O, CPU, and memory, the system determines the optimal plan for execution.

## 8.6 Efficient Storage Encryption Mechanism

GBase 8s currently implements storage encryption in the backend secure storage engine, which is transparent to authorized users and does not affect their frontend operations. There is no loss of functionality. The encryption is applied at the data page level, making it easy to implement and efficient in terms of encryption and decryption. The mechanism of storing ciphertext on physical storage and caching plaintext in memory ensures efficient data retrieval. This ensures that encryption has minimal impact on GBase 8s' original efficient retrieval mechanism, with a performance reduction of no more than 15% caused by storage encryption.

# 9 Interfaces and Management Tools

## 9.1 Development Interfaces

### 9.1.1 ODBC

GBase ODBC is the ODBC driver for GBase 8s, providing access to all ODBC functionality of GBase 8s. GBase ODBC supports ODBC 3.5X level specification (all APIs + Level 2 features). Users can access GBase 8s databases through GBase ODBC driver using the ODBC Data Source Administrator or directly call the GBase ODBC driver. Additionally, visual programming tools like C++ Builder, Visual Studio, etc., can also utilize GBase ODBC for access. GBase ODBC supports Unix, Linux, and Windows platforms, all of which are supported by GBase 8s.

### 9.1.2 JDBC

GBase JDBC is a driver that is compatible with JDBC specifications 3.0 and 4.0 (Type 4). This means it is a pure Java program that conforms to the JDBC 3.0 and 4.0 version specifications and can directly communicate with GBase servers using the GBase protocol. GBase JDBC provides an interface for client applications using the Java programming language to access GBase 8s.

### 9.1.3 ADO.NET

GBase ADO.NET is an interface program that provides convenient, efficient, and secure interaction between .NET applications and GBase databases. It is written in 100% pure C# and inherits from Microsoft ADO.NET classes. Developers can use any .NET development language (C#, VB.NET, F#) to operate on GBase databases through GBase ADO.NET without the need to install the GBase database client.

### 9.1.4 GCI

GBase GCI is a compatible interface standard provided by GBase 8s database, compatible with Oracle OCI. It currently supports 45 Oracle OCI interfaces. Applications can access GBase 8s database by calling GBase GCI.

## 9.2 Programming Languages

GBase 8s supports multiple programming languages, including C, Java, Python, and Go, among others.

## 9.3 Frameworks

GBase 8s supports various development frameworks, including SqlAlchemy (Python ORM framework), XORM (Go), Hibernate, MyBatis, Activity, EFCore (.NET), etc.

## 9.4 Management Tools

GBase 8s provides powerful, feature-rich, and user-friendly graphical management tools to assist database administrators in managing databases.

### 9.4.1 Enterprise Manager

GBase 8s Enterprise Manager provides users with the functionality to access and manage databases.

Through JDBC driver, the Enterprise Manager establishes a connection between the management tool and the database, enabling direct communication with the database

instance using JDBC.

Using GBase 8s Enterprise Manager, you can perform the following tasks:

- View and visually manage databases, tables, views, indexes, stored procedures, functions, and triggers.
- Visual backup and restore of databases.
- Visual viewing of local logs.
- Create and execute SQL statements in the SQL dialog box.
- Select and edit data records in tables.

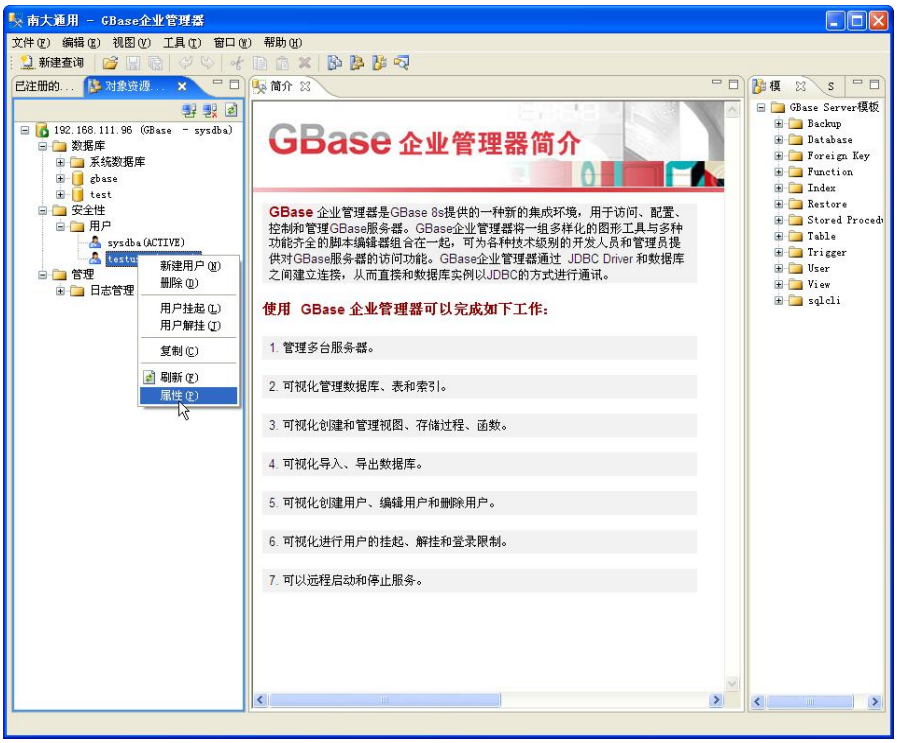



Figure 6.1. GBase 8s Enterprise Manager

# 9.4.2 Security Management Tool

The Security Management Tool is primarily designed for use by security administrators to perform security operations such as subject and object security labeling and security policy definition. The security commands or operations used by security administrators for mandatory access control management can be categorized into five categories: parameter setting commands, browse commands, modification/deletion commands, privilege commands, and security administrator user management.

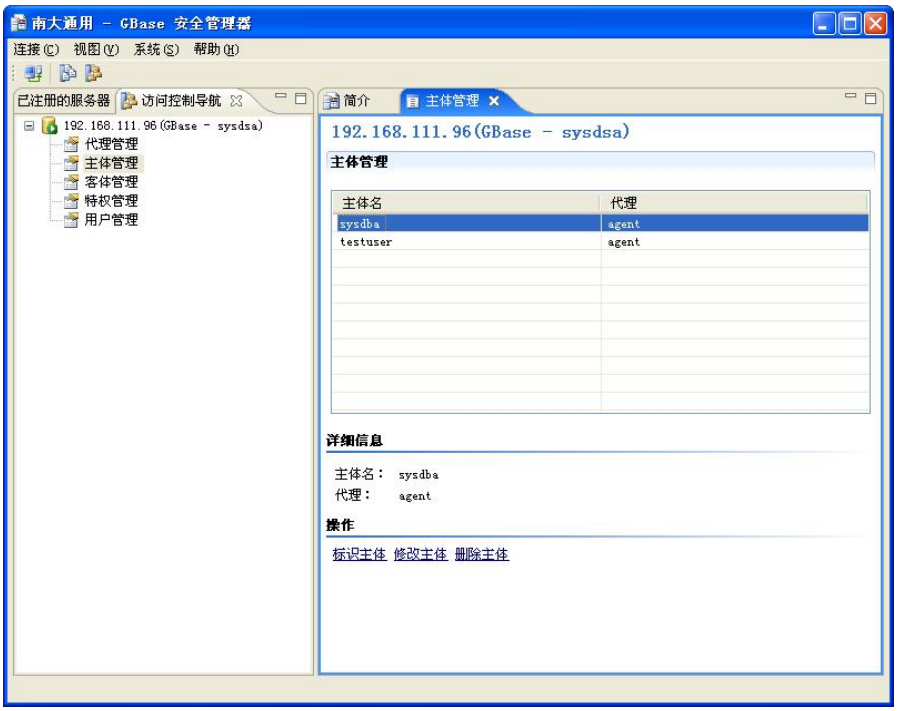



Figure 6.2. GBase 8s Security Manager

## 9.4.3 Audit Management Tool

The Audit Management Tool is mainly used by audit management users to configure audit objects, audit policies, and perform comprehensive query and analysis of audit records.

The audit operations available for audit administrators can be categorized into 4 categories: parameter setting operations, browse operations, modification/deletion operations, and user management operations.

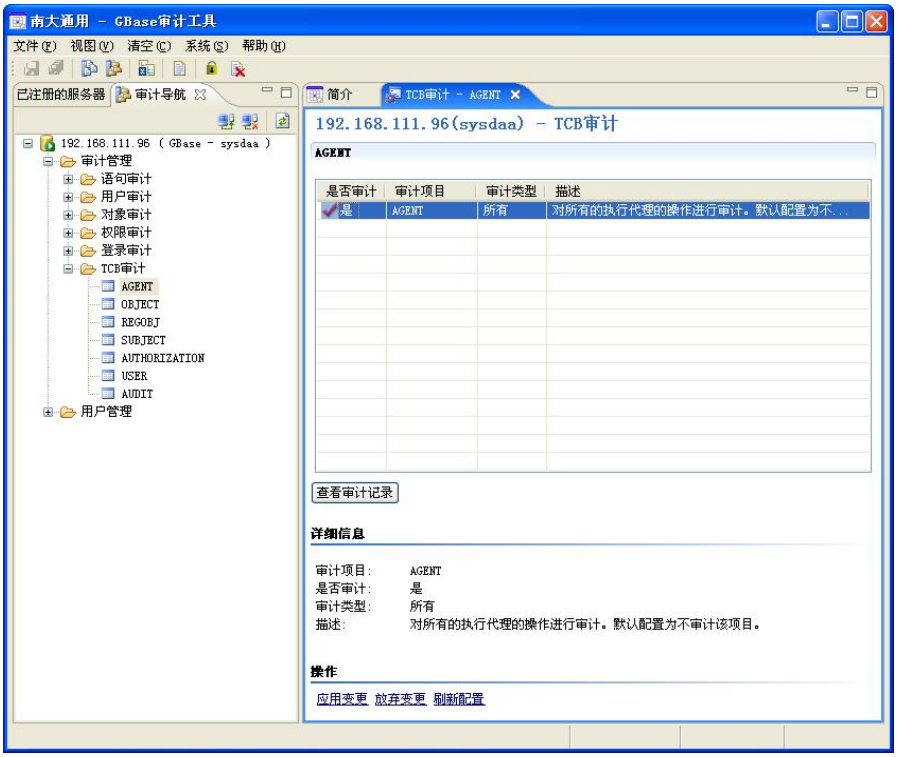



Figure 6.3. GBase 8s Audit Management Tool

## 9.4.4 Backup and Recovery Tool

The Backup and Recovery Tool is used to backup and restore database structures, data files, and database configuration information. The GBase 8s Backup and Recovery Tool provides online backup and recovery methods. During online backup, it can perform full backups, differential backups, and incremental backups based on the situation.

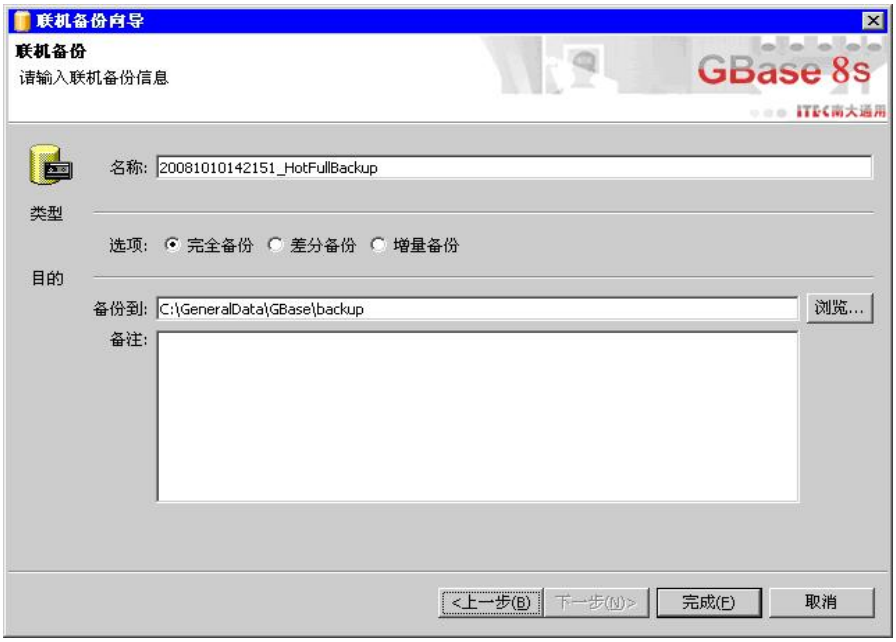



**Figure 6.4.** GBase 8s Backup and Recovery Tool

# 10 Migration Capability

GBase 8s provides a comprehensive solution for risk-free migration from third-party databases to GBase 8s. GBase 8s itself has excellent support for various migration tools, further ensuring the success rate of data migration and improving the efficiency of migration work. GBase 8s supports the following mainstream database migration tools:

- Supports database structure migration tool: MTK
- Supports data migration tool: Kettle
- Supports incremental data migration tools: IBM CDC, Oracle Golden Gate, Informatica

# 11 Technical Services

The technical service levels for GBase 8s V8.8 products are divided into three levels: Level 1 phone support service, Level 2 non-code-level technical service, and Level 3 code-level advanced technical service.

Level 1 (phone support service) is the standard operational service for GBase 8s V8.8. Customers who purchase GBase 8s V8.8 products can use the phone service to consult about product installation, deployment, questions during usage, patch upgrades, and technical issues.

Service Phone: 400-013-9696

Service Email: info@gbase.cn

When customers have questions during product usage, they can contact Level 1 technical support via phone. After analyzing the questions, Level 1 technical support provides feedback on identified issues and transfers issues that cannot be determined to Level 2 technical support for in-depth technical analysis. After conducting in-depth technical analysis, Level 2 technical support provides feedback on issues that do not require code modifications and transfers issues that involve code modifications to Level 3 technical support. Level 3 technical support analyzes the problems and fixes the issues that require code modifications. Within 7-14 days, Level 3 technical support releases temporary patches to Level 1 technical support, who then provides unified feedback to customers. At the same time, Level 3 synchronizes the temporary patches with the development team, who incorporates them into version management.

GBASE®

General Data Technology Co., Ltd.

@GBase

@gbaseglobal



Technical support hotline: 400-013-9696