| Category | Feature/Enhancement | Description |
| --- | --- | --- |
| Scalability and Performance | [Disaggregation of PD to improve scalability (experimental)](https://docs.pingcap.com/tidb/v8.0/pd-microservices) | Placement Driver (PD) contains multiple critical modules to ensure the normal operation of TiDB clusters. As the workload of a cluster increases, the resource consumption of each module in PD also increases, causing mutual interference between these modules and ultimately affecting the overall service quality of the cluster. Starting from v8.0.0, TiDB addresses this issue by splitting the TSO and scheduling modules in PD into independently deployable microservices. This can significantly reduce the mutual interference between modules as the cluster scales. With this architecture, much larger clusters with much larger workloads are now possible. |
| [Bulk DML for much larger transactions (experimental)](https://docs.pingcap.com/tidb/v8.0/system-variables#tidb_dml_type-new-in-v800) | Large batch DML jobs, such as extensive cleanup jobs, joins, or aggregations, can consume a significant amount of memory and have previously been limited at very large scales. Bulk DML (tidb\_dml\_type = "bulk") is a new DML type for handling large batch DML tasks more efficiently while providing transaction guarantees and mitigating OOM issues. This feature differs from import, load, and restore operations when used for data loading. |
| [Acceleration of cluster snapshot restore speed (GA)](https://docs.pingcap.com/tidb/v8.0/br-snapshot-guide#restore-cluster-snapshots) | With this feature, BR can fully leverage the scale advantage of a cluster, enabling all TiKV nodes in the cluster to participate in the preparation step of data restores. This feature can significantly improve the restore speed of large datasets in large-scale clusters. Real-world tests show that this feature can saturate the download bandwidth, with the download speed improving by 8 to 10 times, and the end-to-end restore speed improving by approximately 1.5 to 3 times. |
| Enhance the stability of caching the schema information when there is a massive number of tables (experimental) | SaaS companies using TiDB as the system of record for their multi-tenant applications often need to store a substantial number of tables. In previous versions, handling table counts in the order of a million or more was feasible, but it had the potential to degrade the overall user experience. TiDB v8.0.0 improves the situation with the following enhancements:   * Introduce a new [schema information caching system](https://docs.pingcap.com/tidb/v8.0/system-variables#tidb_schema_cache_size-new-in-v800), incorporating a lazy-loading Least Recently Used (LRU) cache for table metadata and more efficiently managing schema version changes. * Implement a [priority queue](https://docs.pingcap.com/tidb/v8.0/system-variables#tidb_enable_auto_analyze_priority_queue-new-in-v800) for auto analyze, making the process less rigid and enhancing stability across a wider array of tables. |
| DB Operations and Observability | Support monitoring index usage statistics | Proper index design is a crucial prerequisite to maintaining database performance. TiDB v8.0.0 introduces the [INFORMATION\_SCHEMA.TIDB\_INDEX\_USAGE](https://docs.pingcap.com/tidb/v8.0/information-schema-tidb-index-usage) table and the [sys.schema\_unused\_indexes](https://docs.pingcap.com/tidb/v8.0/sys-schema-unused-indexes) view to provide usage statistics of indexes. This feature helps you assess the efficiency of indexes in the database and optimize the index design. |
| Data Migration | TiCDC adds support for [the Simple protocol](https://docs.pingcap.com/tidb/v8.0/ticdc-simple-protocol) | TiCDC introduces a new protocol, the Simple protocol. This protocol provides in-band schema tracking capabilities by embedding table schema information in DDL and BOOTSTRAP events. |
| TiCDC adds support for [the Debezium format protocol](https://docs.pingcap.com/tidb/v8.0/ticdc-debezium) | TiCDC introduces a new protocol, the Debezium protocol. TiCDC can now publish data change events to a Kafka sink using a protocol that generates Debezium style messages. |

| Category | Feature/Enhancement | Description |
| --- | --- | --- |
| Scalability and Performance | [Cross-database SQL binding](https://docs.pingcap.com/tidb/v7.6/sql-plan-management#cross-database-binding) | When managing hundreds of databases with the same schema, it is often necessary to apply SQL bindings across these databases. For example, in SaaS or PaaS data platforms, each user typically operates separate databases with the same schema and runs similar SQL queries on them. In this case, it is impractical to bind SQL for each database one by one. TiDB v7.6.0 introduces cross-database SQL bindings that enable matching bindings across all schema-equivalent databases. |
| [Achieve up to 10 times faster for snapshot restore (experimental)](https://docs.pingcap.com/tidb/v7.6/br-snapshot-guide#restore-cluster-snapshots) | BR v7.6.0 introduces an experimental coarse-grained Region scatter algorithm to accelerate snapshot restores for clusters. In clusters with many TiKV nodes, this algorithm significantly improves cluster resource efficiency by more evenly distributing load across nodes and better utilizing per-node network bandwidth. In several real-world cases, this improvement accelerates restore process by about up to 10 times. |
| [Achieve up to 10 times faster for creating tables in batch (experimental)](https://docs.pingcap.com/tidb/v7.6/ddl-v2) | With the implementation of the new DDL architecture in v7.6.0, the performance of batch table creation has witnessed a remarkable improvement, up to 10 times faster. This substantial enhancement drastically reduces the time needed for creating numerous tables. This acceleration is particularly noteworthy in SaaS scenarios, where the prevalence of high volumes of tables, ranging from tens to hundreds of thousands, is a common challenge. |
| [Use Active PD Followers to enhance PD's Region information query service (experimental)](https://docs.pingcap.com/tidb/v7.6/tune-region-performance#use-the-active-pd-follower-feature-to-enhance-the-scalability-of-pds-region-information-query-service) | TiDB v7.6.0 introduces an experimental feature "Active PD Follower", which allows PD followers to provide Region information query services. This feature improves the capability of the PD cluster to handle GetRegion and ScanRegions requests in clusters with a large number of TiDB nodes and Regions, thereby reducing the CPU pressure on the PD leader. |
| Reliability and Availability | [Support TiProxy (experimental)](https://docs.pingcap.com/tidb/v7.6/tiproxy-overview) | Full support for the TiProxy service, easily deployable via deployment tooling, to manage and maintain connections to TiDB so that they live through rolling restarts, upgrades, or scaling events. |
| [Data Migration (DM) officially supports MySQL 8.0 (GA)](https://docs.pingcap.com/tidb/v7.6/dm-compatibility-catalog) | Previously, using DM to migrate data from MySQL 8.0 is an experimental feature and is not available for production environments. TiDB v7.6.0 enhances the stability and compatibility of this feature to help you smoothly and quickly migrate data from MySQL 8.0 to TiDB in production environments. In v7.6.0, this feature becomes generally available (GA). |