



APUSIC
固若长城
睿比世界

Product Overview

金蝶Apusic分布式缓存 V2.0.4

版权声明

本文档所涉及的软件著作权、版权等知识产权已依法进行了注册，由金蝶天燕云计算股份有限公司合法拥有。受《中华人民共和国著作权法》《计算机软件保护条例》《知识产权保护条例》和相关国际版权条约、法律、法规以及其它知识产权法律和条约的保护。未经授权许可，不得非法使用。

免责声明

本文档包含的版权信息由金蝶天燕云计算股份有限公司合法拥有，受法律的保护，金蝶天燕云计算股份有限公司对本文档可能涉及到的非金蝶天燕云计算股份有限公司的信息不承担任何责任。在法律允许的范围内，您可以查阅并仅能够在《中华人民共和国著作权法》规定的合法范围内复制和打印本文档。任何单位和个人未经金蝶天燕云计算股份有限公司书面授权许可，不得使用、修改、再发布本文档的任何部分和内容，否则将被视为侵权，金蝶天燕云计算股份有限公司有依法追究其责任的权利。

本文档如有更新，不另行通知。对本文档中的问题您可向金蝶天燕云计算股份有限公司告知或查询。未经本公司明确授予的任何权利均予保留。

商标声明

 是深圳市金蝶天燕云计算股份有限公司向中华人民共和国国家商标局申请注册的注册商标，注册商标专用权由金蝶天燕合法拥有，受法律保护。未经金蝶天燕的书面许可，任何单位及个人不得以任何方式或理由对该商标的任何部分进行使用、复制、修改、传播、抄录或与其它产品捆绑使用销售。凡侵犯金蝶天燕商标权的，金蝶天燕将依法追究其法律责任。本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

目录

- 1 Preface
 - 1.1 Target Audience
 - 1.2 Related Documentation
 - 1.3 Technical Support
- 2 Product Introduction
- 3 Core Features
 - 3.1 High-Performance Cache Engine
 - 3.2 High Availability Architecture
 - 3.3 Security Enhancement
 - 3.4 Operations Management
 - 3.5 Development Support
- 4 Technical Features
 - 4.1 High Performance
 - 4.2 High Reliability
 - 4.3 High Security
 - 4.4 High Compatibility
- 5 Application Scenarios
 - 5.1 Centralized Session Storage for Application Servers
 - 5.2 Hot Data Caching
 - 5.3 Distributed Lock
 - 5.4 Lightweight Message Queue
 - 5.5 Counter and Interface Rate Limiting
 - 5.6 Time-Limited Business

1 Preface

This document provides an introduction to Apusic In-Memory Data Cache (AMDC) V2.0.4, helping users quickly understand the product's features, technical characteristics, and typical application scenarios.

1.1 Target Audience

This document is intended for IT information business managers, R&D managers, software project managers, software architects, and operations engineers.

1.2 Related Documentation

For more information about AMDC V2.0.4, please refer to the following AMDC V2.0.4 product manual documentation set:

No.	Document	Description
1	Apusic In-Memory Data Cache V2.0.4 Quick Start Guide	A brief introduction on how to quickly get started with AMDC.
2	Apusic In-Memory Data Cache V2.0.4 Installation Guide	Detailed introduction on how to install AMDC on various operating systems, AMDC service start/stop operations, and product registration process.
3	Apusic In-Memory Data Cache V2.0.4 Core User Manual	Detailed introduction on how to use, configure, and manage AMDC features and supporting tools.
4	Apusic In-Memory Data Cache V2.0.4 Console User Manual	Detailed instructions on using and operating AMDC console features.
5	Apusic In-Memory Data Cache V2.0.4 Development Guide	Detailed instructions for developing AMDC client applications using various programming languages.
6	Apusic In-Memory Data Cache V2.0.4 Migration Guide	Detailed instructions for migrating AMDC from historical versions to V2.0.4, as well as migrating from Redis to AMDC.
7	Apusic In-Memory Data Cache V2.0.4 Operations Guide	Detailed instructions for AMDC monitoring, operations, security hardening, and other operational procedures.
8	Apusic In-Memory Data Cache V2.0.4 Performance Tuning Guide	Detailed instructions for AMDC performance tuning.

1.3 Technical Support

AMDC products provide comprehensive technical support services. You can obtain technical support through the following channels:

- Website: www.apusic.com
- Phone: 400-855-5800
- Email: support@apusic.com
- Kingdee Cloud Community: <https://vip.kingdee.com/?productId=73&productLineId=14&lang=zh-CN>

When requesting technical support, please provide the following information:

1. Your name
2. Company information and contact details
3. Operating system and version
4. Product version number
5. Detailed information including logs, screenshots, etc. of any exceptions or errors

2 Product Introduction

Apusic In-Memory Data Cache Software (Apusic In-Memory Data Cache, abbreviated as AMDC) is an enterprise-level high-performance distributed cache middleware that is 100% compatible with Redis protocol and development ecosystem, enabling seamless migration without modifying existing business applications. AMDC features core capabilities including high-performance millisecond-level response, high availability with multi-replica fault tolerance, high scalability with elastic expansion, and refined web-based visual management, providing secure and reliable high-frequency data access support for critical business in high-concurrency and distributed scenarios, reducing database pressure and improving overall system throughput.

AMDC is built on Kingdee Apusic's years of middleware technology accumulation, with fully autonomous and controllable underlying architecture to upper-level functions, meeting the localization replacement requirements in information technology innovation environments, while providing enterprise-level technical support and service guarantees.

3 Core Features

3.1 High-Performance Cache Engine

- **Ultra-high throughput:** Multi-threaded network I/O architecture and lock-free concurrency model, single-node throughput up to 140,000 QPS
- **Microsecond-level response:** Pure memory operations combined with efficient data structures, achieving microsecond-level operation response
- **Rich data structures:** Complete support for Redis core data structures including String, Hash, List, Set, Sorted Set, Geo, HyperLogLog, Stream, etc.
- **Flexible cache policies:** Supports multiple memory eviction policies and key expiration mechanisms such as LRU, LFU, TTL, etc.

3.2 High Availability Architecture

- **Master-slave replication:** Implements data redundancy and read-write separation, improving system availability
- **Sentinel mode:** Provides high availability capabilities including automatic fault detection, failover, and configuration propagation
- **Cluster mode:** Supports data sharding storage, elastic scaling, automatic failover, meeting large-scale business requirements
- **Dual persistence mechanism:** Supports both RDB snapshot and AOF log persistence methods, can be used individually or in combination, balancing data persistence efficiency and integrity

3.3 Security Enhancement

- **Access control:** Supports password authentication and ACL fine-grained permission control
- **Transport encryption:** Supports SSL/TLS mutual authentication
- **Audit logging:** Records complete operation logs to meet compliance requirements
- **Command security:** Supports renaming or disabling dangerous commands

3.4 Operations Management

- **Visual management:** Provides Web console, supports centralized unified operations management for multi-version cache core instances
- **Cluster management:** Provides complete toolchain for cluster creation, inspection, repair, and scaling
- **Performance monitoring:** Real-time monitoring of key metrics such as QPS, response time, memory usage, etc.
- **Slow query analysis:** Automatically records and analyzes slow query commands to assist performance tuning
- **Observability:** Built-in Prometheus support, providing rich runtime metrics and standard Grafana Dashboard

3.5 Development Support

- **Protocol compatibility:** Fully compatible with RESP2/RESP3 protocols, supports seamless integration with Redis clients
- **Multi-language support:** Compatible with mainstream client libraries such as Jedis, Lettuce, redis-py, go-redis, etc.
- **Lua scripting:** Built-in Lua interpreter, supports atomic execution of complex logic
- **Complete toolchain:** Provides CLI tools, benchmark tools, file repair tools, configuration conversion tools, etc.

4 Technical Features

4.1 High Performance

- Network layer multi-threaded concurrent architecture
- Single-threaded command execution ensures atomicity
- Memory-optimized data structure design
- Zero-copy technology reduces memory overhead

4.2 High Reliability

- Multi-replica data redundancy
- Decentralized cluster architecture
- Automatic fault detection and switching
- Data consistency guarantee mechanism

4.3 High Security

- ACL fine-grained permission control
- SSL/TLS encrypted transmission

4.4 High Compatibility

- Complete Redis protocol compatibility
- Mainstream programming language client support
- Standardized API interfaces
- Rich tool ecosystem

5 Application Scenarios

5.1 Centralized Session Storage for Application Servers

Provides unified Session storage capabilities for application server (such as Tomcat, AAS, etc.) cluster deployments, solving the problem of isolated Session data across cluster nodes and session state loss when users access across nodes, achieving global session sharing and unified management.

5.2 Hot Data Caching

Serves as a "front-end buffer layer" for databases, storing frequently accessed hot data in applications, reducing high-frequency query pressure on databases, and preventing database overload due to high concurrent requests. Shortens data access path, reducing response latency from millisecond-level at the database to microsecond-level, improving overall system throughput and user experience.

5.3 Distributed Lock

Provides cross-service, cross-node mutually exclusive access control for concurrent operations on shared resources in distributed architectures. By using distributed cache, concurrent conflicts in distributed environments can be effectively avoided, ensuring that the same resource is operated by only one service instance at the same time, guaranteeing data consistency.

5.4 Lightweight Message Queue

Can be used as a message queue to implement asynchronous communication between different services. By decoupling synchronous dependencies between services, it prevents the entire link from being blocked due to a service failure. Supports peak shaving and valley filling, buffering message peaks in high-concurrency scenarios to improve system resilience and stability.

5.5 Counter and Interface Rate Limiting

Based on atomic operations, achieves precise counting of target objects (such as page views, clicks, login counts), or limits request frequency for interfaces and users. Meets real-time counting requirements in high-concurrency scenarios (ensuring accurate counting without concurrent errors); prevents interfaces from being overwhelmed by malicious requests or traffic spikes, protecting backend service resources and ensuring system availability.

5.6 Time-Limited Business

Implements various time-limited business logic through TTL mechanism, completing operations within specified time or automatically expiring. Suitable for scenarios such as verification codes, coupons, limited-time activities, etc.

全国统一服务热线
4008-555-800



金蝶天燕云计算股份有限公司(简称“金蝶天燕云”)成立于2000年，前身为“金蝶中间件公司”，是金蝶集团旗下新一代软件基础云平台服务商，云计算国家标准制定企业，国家信创产业核心软件企业。金蝶天燕是国家863重点研发计划与核高基重大专项承接企业，也是“两网一站四库十二金”国家重点工程的基础平台提供商，产品广泛应用于政府、军工、金融、能源等关键行业，累计服务客户总数超过10万家。



云计算国家标准制定企业
金蝶集团旗下基础软件企业
信息技术应用创新核心企业
官网：www.apusic.com

