100 XP

How Azure Cache for Redis works

7 minutes

Here, you'll review the features of Azure Cache for Redis works and how they work together to help improve app functionality. This knowledge will help you decide whether you can use it in your organization.

Azure Cache for Redis provides a number of use-cases that help improve the performance and scalability of apps that rely heavily on back-end data stores. You'll learn what the following Azure Cache for Redis use-cases provide:

- Distributed cache
- Session store
- Message broker
- Cloud migration

Distributed cache

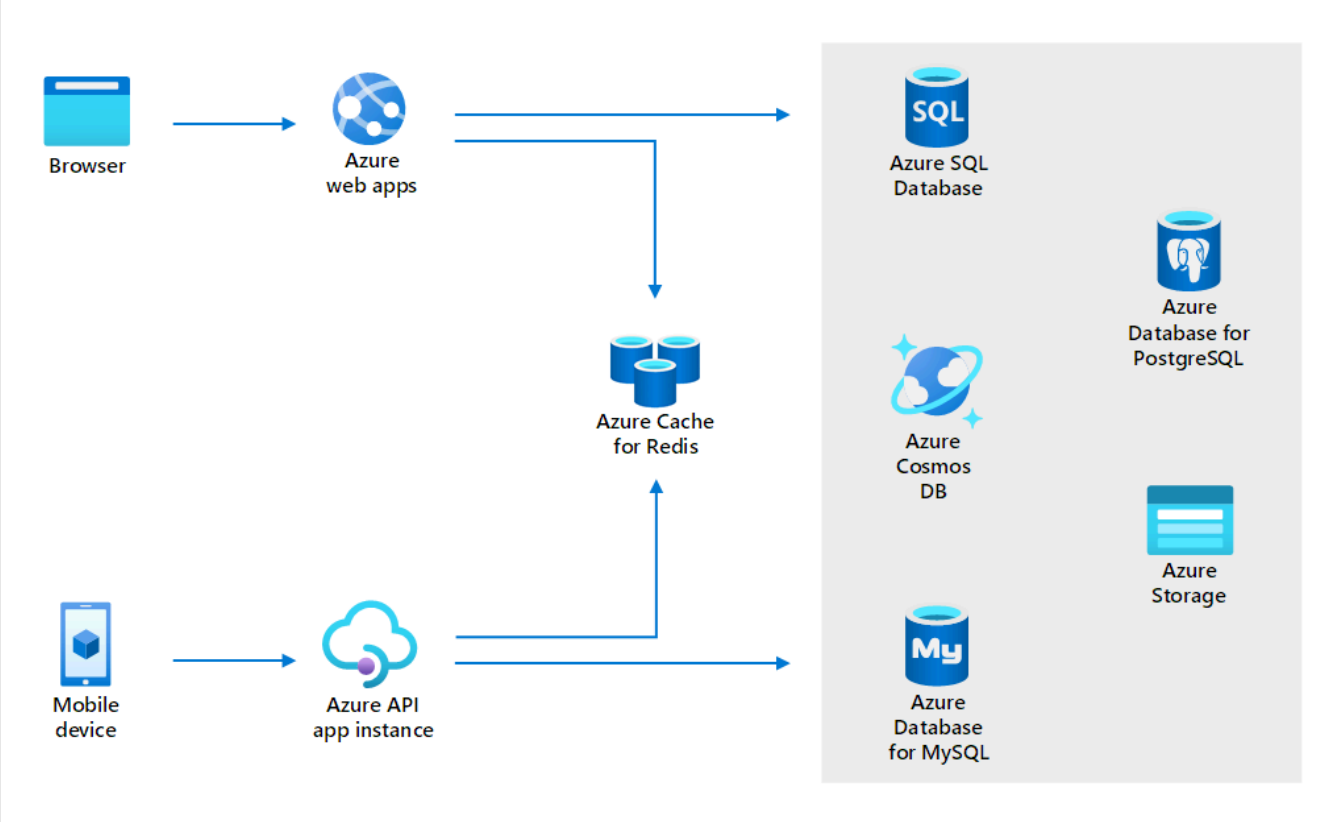
The distributed cache use-case in Azure Cache for Redis helps improve your apps' response times by copying frequently-accessed data to a cache. This cache has lower latency and provides for higher throughput than the primary datastore. The distributed cache feature:

- Accelerates application responsiveness.
- Helps reduce load on primary datastores and compute resources.
- Integrates with many Azure databases, including Azure SQL and Azure Cosmos DB.

You can use distributed cache to:

- Manage spikes in traffic.
- Cache and provide commonly accessed data to users.
- Help reduce compute load on your databases.
- Locate content geographically closer to users.
- Provide for output caching.

As indicated in the following graphic, Azure Cache for Redis can help improve performance in apps that interface with many database solutions, including Azure SQL Database, Azure Cosmos DB, and Azure Database for MySQL.



Session store

Your session-oriented apps require the ability to store and access temporary session data when users sign in and remain active on your apps. The session store use-case in Azure Cache for Redis:

- Manages up to hundreds of thousands of simultaneous users.
- Makes data-replication options available to help provide for maximum reliability.
- Helps reduce costs, as it's typically more cost-effective and scalable than alternative database or storage options.

You can use session store to:

- Help facilitate eCommerce shopping carts.
- Store user cookies.
- Maintain user login and session state data.
- Enable IoT telemetry.

Message broker

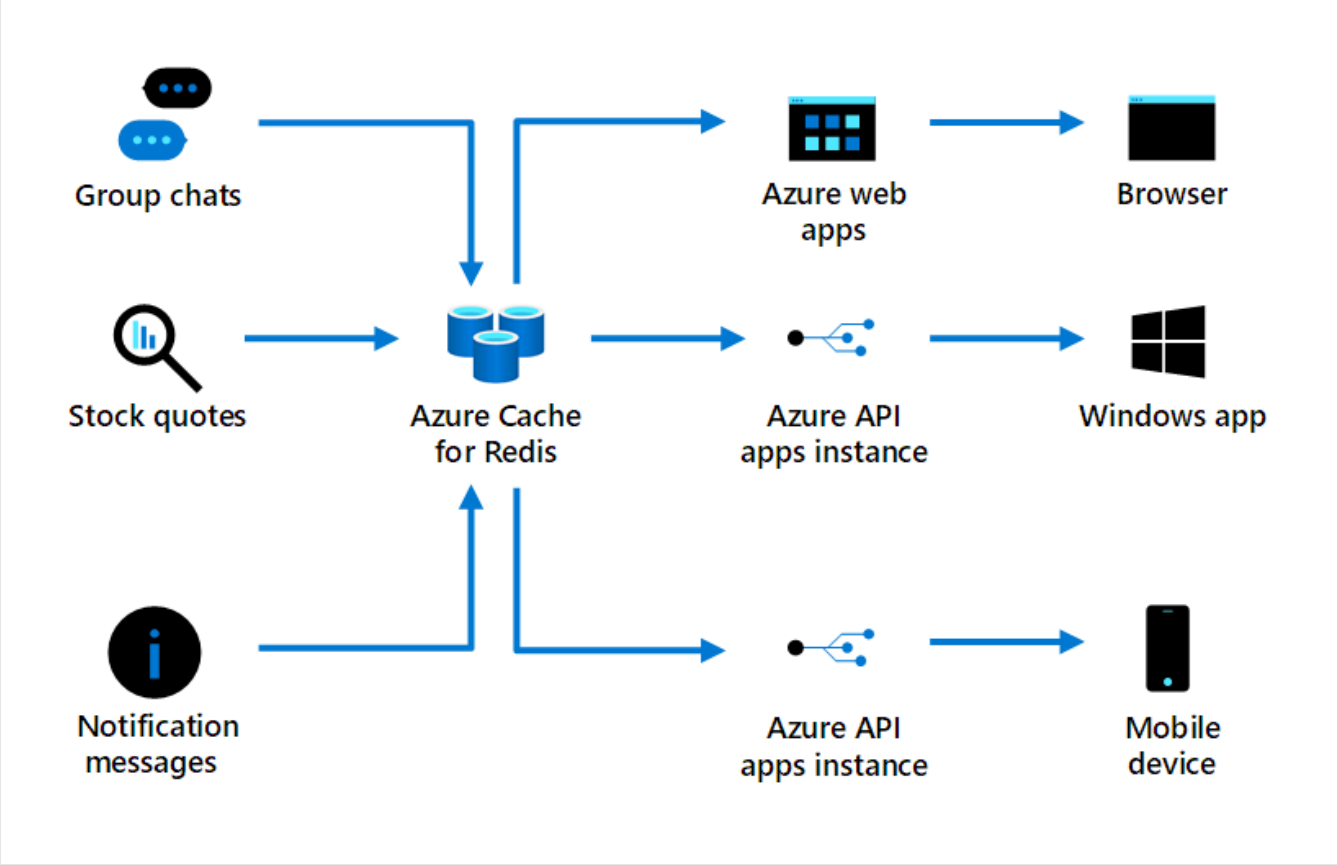
Apps built on microservices often need to asynchronously communicate. Azure Cache for Redis can implement a publish/subscribe or queue architecture that can help enable fast and reliable communication between these microservices. The Azure Cache for Redis message broker:

- Provides a temporary data store with minimal overhead and cost.
- Supports TLS encryption for data in transit.
- Provides network isolation for secure communication between your services.

You can use the message broker use-case to:

- Publish news, financial data, or application updates to users.
- Manage chat messages.
- Enable communication between microservices.

In the following graphic, a number of group chats, notifications, and stock quotes are occurring. They're connected to the message-broker feature in Azure Cache for Redis. This feature, in turn, connects to Azure API Apps instances and Azure web apps. These elements provide access to the group chats, notifications, and stock quotes for connected client devices.



Cloud migration

If you're moving from an on-premises cache to a managed service, a critical factor is how to move content to the managed service. Azure Cache for Redis helps migrate data to the cloud and also:

- Enables both import and export of RDB files.
- Provides compatibility with open-source Redis to help simplify migration.

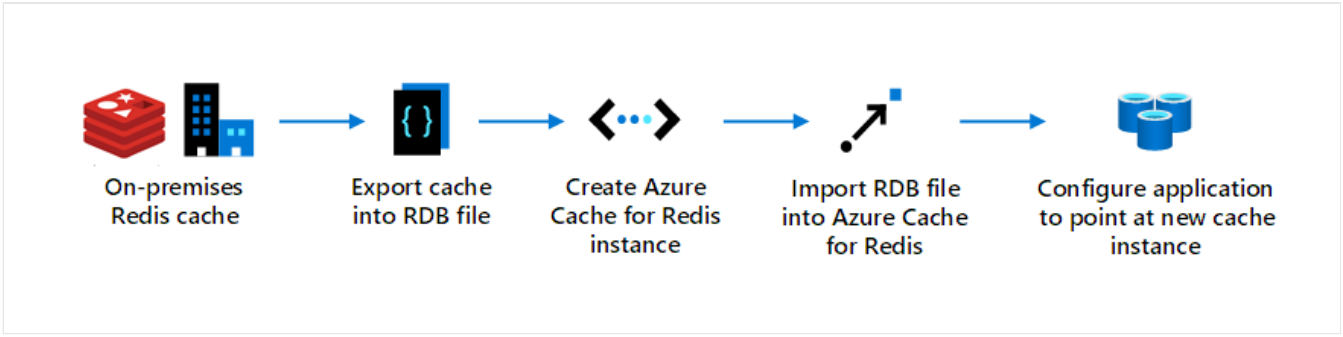
- Provides a fully managed service that manages:
  - Patching
  - Updates
  - Provisioning
  - Scaling
  - Setup

You can use cloud-migration in Azure Cache for Redis to:

- Migrate your apps from your on-premises environment to the cloud.
- Help modernize your current infrastructure as a service (IaaS) apps through the benefits of platform as a service (PaaS) services.

A typical process proceeds as follows:

1. From an existing on-premises Redis cache, you export the cache to an RDB file.
2. You create an Azure Cache for Redis instance.
3. Next, you import the RDB into this instance.
4. Finally, you configure your new application to point to your Azure Cache for Redis instance.



Next unit: When to use Azure Cache for Redis

Continue >