



We use cookies to improve your experience on our websites and for advertising.

[Privacy Statement](#)

Accept all

Manage cookies



Get started developing for the cloud

Show Me

[ASP.NET Forums](#) / [General ASP.NET](#) / [ASP.NET Core](#) / [What's the difference between Distributed Memory Cache/Distributed SQL...](#)

What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distributed Redis Cache? [Answered]

13 replies

Last post Nov 20, 2019 08:33 AM by [mywatermelon](#)

What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distributed R...

Nov 15, 2019 06:57 AM | [mywatermelon](#)

I made a chatroom app and now I need to save and get the message chatted from the database.

In my opinion, I should use a cache to prevent saving/loading the data from the database frequently.

I am going to use Redis first and soon found there are three ways to achieve this in(<https://docs.microsoft.com/en-us/aspnet/core/performance/caching/distributed?view=aspnetcore-3.0> (<https://docs.microsoft.com/en-us/aspnet/core/performance/caching/distributed?view=aspnetcore-3.0>)): Distributed Memory Cache/Distributed SQL Server Cache/Distributed Redis Cache

I don't know which one should I use for I don't know What's the difference between them.

Now the database I am using is the SQL Server, it seems I should use the Distributed SQL Server Cache for better, right?

What's more, I am confused by the Distributed Memory Cache and In-memory Cache. I don't know the difference between them also.

Would you please help me and give me a suggestion? Thank you.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 15, 2019 04:12 PM | bruce (sqlwork.com)

to pick a cache, your main decision is if a distributed cache is required.

a in-memory cache is a cache system based on 1 web server. if you have a web farm and use a in-memory cache on each server, every server pays for every cache miss. say you decide to cache database request a, each server must process the database query request at least once to cache it.

a distributed cache is designed for a web farm. a distributed cache a common cache in front of a web farm. each farm does not need its own cache, it uses the distributed cache. again if you decide to cache database request a, then the first request for database request a caches for all servers.

distributed caches need a backing store (as they are really a distributed database). this can be in-memory like redis or a database like sql server cache. to pick between redis and sql server cache will depend on the type of caching you need, and the type of cache queries.

distributed memory databases are hard to design. The first time the Microsoft sql server team tried to write one, it was slower than standard sql server and they cancelled the project.

if you have a single web server any will work, but in-memory is probably sufficient. if you have a web farm, then you need to decide.

you are using sqlsever I'd tune the queries before I added a distributed cache. I'd only look at distributed cache if I planed on over 1000 concurrent users.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 16, 2019 02:41 PM | mywatermelon

Thanks for telling me these.

According to my understanding, the in-memory cache can only get/set by the same server while the distributed cache can get/set by any other server, right?

Anyhow, here is the last question:

I have searched more about the Redis and found that the official version can only support the Linux system. Now my server is windows server 2012 that can not install the WSL. Although Microsoft has made a Redis of windows version while it doesn't update for a long time. The windows version still is 3.0+ while the Linux version is 5.0+.

Now I doubt that whether I am going to use Redis in a Windows system is right. If not, which distributed memory database should I use in the windows system? Thank you.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 16, 2019 10:06 PM | bruce (sqlwork.com)

which you pick depends on your requirements. what type of cache queries will you make? how complex are the cache queries? what are the latency requirements? what are the cache stale requirements? what are the expected concurrent request counts? how many web servers are in the farm?

as the distributed server cache usually runs on its own dedicated servers, if you pick rdis, I'd just add a couple linux servers or docker containers to support it.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 12:48 AM | EnenDaveyBoy

i might be wrong, but when i was looking at cache i had a similar issue when reading that article.

However i think its more down to a badly named interface.

IDistributedCache is just an interface that has easy implementation with Memory Cache, SQL Server cache, and Redis cache, and has very little to do with Distrubuted Cache that bruce references in his first post.

I am not sure you need cache for a chat app, you set signalr, to send and receive the message, you use the UI to show the chat (cache it) and manage the chat, and if you need to store the chat for things like later reference it when you send a message get signalr hub to save it in a database.

the main reason for cache is if it were going to call the chat history often

in memory,you will loose the chat if the server is reset, and will have limited room compared to databases.

sql server has a licencess fee, so will have more costs

redis has a free licensee, so will be cheaper/free depending on your hosting

the rest is down to if you already know sql server or redis and how much time to want to spend in learning them,and if they offer what you want

you could also use cosmos db:(there is also a cost to this)

<https://github.com/Azure/Microsoft.Extensions.Caching.Cosmos>

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 06:05 AM | mywatermelon

Thanks for your suggestion.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 06:18 AM | mywatermelon

I made the front-end by the Blazor server-side.

What I want to do is to call the chat history often that you said.

As we know, Blazor server-side can not store anything locally.

In my idea, when the user opens the chat dialog to someone(for example, user A), the program will load the chat history to user A before from the database and set it to the cache(for example Redis).

All the new chat history will add to the cache.

And also, there is a task to insert the new chat history in the cache to the database every minute.

Is that the best way? Please give me a suggestion. Thank you.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 06:23 AM | mywatermelon

The chat room is only made by me but not a company. I have heard that the Cache of SQLServer will cost much. I don't want to cost it more. It seems I have to use the Redis by the docker in a windows system as what @bruce said.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 03:36 PM | EnenDaveyBoy

whats best is upto you, and only you, its likely you'll get it wrong and change it, don't worry thats part of the course.

if you store things in redis by default it will only last 3 hours, then it will be removed (you can change the the time if needed), thats the nature of cache.

so u will still need a database, there are free options, like sql express or mysql

i have never used blazer so have no idea how it works.

it all depends on what the chat app is for, how it will be used,who will use it and on what platform.

open chat //if history is required call the database to retrieve history (only if this is going to be called by alot of people within 3 hours would i cache it(it saves money and development time)

chat loads into the viewer(the browser), and is technically cached thereas it stored in memory or the dom

a message is created and send // signalr sends the message to the hub, the hub records the message in the database and sends the message to all the people in the chat

possible cleanup// if there are too many messages in the browser, you might need to remove some

so its debateable if you need external cache at all

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 17, 2019 09:38 PM | bruce (sqlwork.com)

caching chat history is a poor design (it doesn't really support any staleness). a database or simple table store is a much better design. also moving the chat to the client removes any need for a cache.

in a chat application, typically you cache the online user list and the servers they are connected to (for fast message delivery).

some rules for a typical large chat design:

the should be 1 chat connection server per 10,000 current connected users (typical max connection limit)
a chat application service per 100,000 current users.
the database should be portioned to about 1 database server per 1,000,000 subscribers.
a in memory cache should be used for the partitioning of client to database (assuming connection failover)
a distributed cache should be used for the online connection list and the server the user is connected to.

in your case you seem to be doing a simple limited chat with blazor, signal/r and a single server. (maybe a couple hundred concurrent connections).

you should use a database / table store for the chat history. Your chat history query should be under 50ms.
you should use a database / table store for the chat account database. The account query should be under 40ms.
you should use a in memory cache for the connected user list (needs to be distributed if you go above one server, also requires a bus delivery system for signal/r).

you should really move the chat client to browser. your current architecture is a small javascript client that connects to signal/r messages from the server and sends messages to the server blazor engine to process.

when a signal/r message is received by the browsers javascript, it calls back to the server with the message to the blazor engine. the blazor engine process the message to render the html. the blazor engine sends the html back to the client, then javascript then re-renders the html.

if the user sends a message, the data is posted to the blazor engine, the blazor engine calls back to the browser to have the javascript send the signal/r message. the blazor engine can also create new html to send back to the javascript client to render.

removing this double hop per chat message would be better than any cache you might do.

note: the blazor engine is a sandbox. it has no dom, file, or network access. it always calls javascript to access the network, filesystem or update the dom. if the blazor engine is hosted server side, a winsocket is used to interop to the browsers javascript. for example the WebClient in blazor is just an javascript interop call to the browsers XMLHttpRequest object.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 18, 2019 01:49 PM | mywatermelon

I don't want to store them in Redis directly. Redis is just to read/write the chat history frequently. When the data be removed from Redis after 3 hours as you said. The program will read and write to Redis again automatically. I won't need Redis only but also need a database.

Blazor server-side is use SignalR to transfer data automatically(<https://docs.microsoft.com/en-us/aspnet/core/blazor/hosting-models?view=aspnetcore-3.0> (<https://docs.microsoft.com/en-us/aspnet/core/blazor/hosting-models?view=aspnetcore-3.0>)).

Although I think it is an unrealized dream that there are thousands of users, I don't want just make a toy but learn more about it(such as how to deal with lots of people access it at the same time as above).

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 18, 2019 02:00 PM | mywatermelon

Thanks for helping me and I have learned more from what you said.

Honestly, my blazor program won't access by many people at the same time.

However, I just want to learn more about how to deal with many chats connection at the same time. I don't want to just make a lovely toy. Aha.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 18, 2019 05:05 PM | bruce (sqlwork.com)

server side blazor uses signal/r to talk to the supporting blazor javascript client. if you use signal/r for chat, it is the javascript client that gets the messages. The server blazor code is just a called by the hub in response to client messages.

if this is a learning exercise, then you should learn the difference between scale-out and scale-up. SQLServer is a scale-up technology. while redis is scale-out. scale-out requires keeping distributed nodes in sync. scale-out performance is often measured in the performance gain each node supplies, and the max number of nodes supported. redis scales up to about 1000 nodes.

Re: What's the difference between Distributed Memory Cache/Distributed SQL Server Cache/Distribut...

Nov 20, 2019 08:33 AM | mywatermelon

Thank you.

This site is managed for Microsoft by Neudesic, LLC. | © 2021 Microsoft. All rights reserved.

| [Change cookie preferences](#)