# The pros and cons of the CQRS architecture pattern

*https://www.redhat.com/en/blog/pros-and-cons-cqrs*



Having a grasp of common architectural patterns is essential to designing software architecture at scale. Using them not only saves time but also ensures a reliable implementation of your design. There?s no need to reinvent the wheel when there?s an architectural pattern available that applies to an architecture you?re developing.

The following is a brief overview of the CQRS architectural pattern.



The Command Query Responsibility Segregation (CQRS) pattern separates a service?s write tasks from its read tasks. While reading and writing to the same database is acceptable for small applications, distributed applications operating at web-scale require a segmented approach. Typically there?s more read activity than write activity. Also, read activity is immutable. Thus, replicas dedicated to reading data can be spread out over a variety of geolocations. This approach allows users to get the data that closest to them. The result is a more efficient enterprise application. Separating write from read activity also allows Enterprise Architects to create a variety of read databases, with each database optimized for a specific purpose.

Pros

Cons



CQRS is a popular architecture pattern because it addresses a common problem to most enterprise applications. Separating write behavior from read behavior, which the essence of the CQRS architectural pattern, provides stability and scalability to enterprise applications while also improving overall performance. Choose it for those times when contention over read and write access

overwhelms your systems and you're willing to make the investment that's necessary to address the issue. For a deeper dive, read our illustrated guide to CQRS.

Bob Reselman is a nationally known software developer, system architect, industry analyst, and technical writer/journalist. Over a career that spans 30 years, Bob has worked for companies such as Gateway, Cap Gemini, The Los Angeles Weekly, Edmunds.com and the Academy of Recording Arts and Sciences, to name a few. He has held roles with significant responsibility, including but not limited to, Platform Architect (Consumer) at Gateway, Principal Consultant with Cap Gemini and CTO at the international trade finance company, ItFex.