



## A BRIEF INTRODUCTION



Big data, agile development, and cloud computing are driving new requirements for database management systems. These requirements are in turn driving the next phase of growth in the database industry, mirroring the evolution of the OLAP industry. This document describes this evolution, the new application workload, and how MongoDB is uniquely suited to address these challenges.



10gen

## DATABASE EVOLUTION

As the database market evolves, NoSQL space has emerged as a pillar of enterprise data architecture, providing tools essential to the success of modern IT organizations.

During the past 30 years, Relational Database Management Systems (RDBMS) provided essentially the only option for persistent application data storage. When originally conceived, RDBMS offered increased flexibility over the individually built custom databases of the past, and enabled great leaps forward in productivity through the introduction of a standard data modeling and query language.

### The Emergence of OLAP

Ten years ago, however, the industry-wide expansion of data collection, data storage and large-scale, databases created the need for technologies more suited to analytical workloads. These workloads, characterized by queries that accessed every record in the database, ran too slowly and impacted the performance of primary transaction processing.

By organizing data into columns instead of rows, Online Analytical Processing (OLAP) provided a way for these analytical workloads to run many times faster, and therefore free up resources for the RDBMS to continue processing transactions quickly. It was a leap forward that provided increased capacity – a preview of the evolution happening today thanks to NoSQL.

As with analytical processing in the past, application owners today are discovering that modern application data models and workloads do not fit well with the design of the relational database. NoSQL represents a significant paradigm shift, applying novel methods to overcome the limitations of the RDBMS, leaving the RDBMS to excel at its core functionality: transaction processing.



## THE CHANGING WORKLOAD

Today's new workloads and demanding pace of product release schedules create a need for new database technologies. These new requirements call for a database that is optimized for:

- Big data with high operation rates
- Agile development
- Cloud computing

### Big data with high operation rates

The volume of data businesses store about users, objects, products, and events is exploding, outpacing the advancement of processing power and storage capacity. At the same time this data grows, data is accessed more frequently, and with more granularity. As applications become more interactive, networked and social, they drive more requests to the database. Rendering a single web page or answering a single API request can now take tens or hundreds of database requests, and this trend will only continue to expand. In order to keep up with throughput requirements, the industry must find new ways of managing data.

**Agile development**

The way we construct software has changed dramatically since the RDBMS was originally created. Engineers today utilize iterative development methodologies, which aim for continuous deployment and short development cycles. In order to sustain this pattern of development, an application’s data store must be very flexible.

An RDBMS requires the definition of a schema before you can add data. This fits poorly with agile development approaches, because each time you complete new features, the schema of your database often needs to change. If the database is large, this means a very slow process. If application releases are frequent, scheduling schema migrations and maintenance windows simply becomes impractical.

While mapping data from today’s object-oriented programming languages to a relational model is feasible, it requires significant effort and is contrary to the rapid development philosophy of agile software development methodologies.

**Cloud computing**

The move to cloud computing is one of the most influential trends in enterprise computing. Whether public or private, when you deploy your applications, it is likely to be into a virtualized, cloud-based environment. Developers no longer engineer complex high-end hardware platforms to support applications.

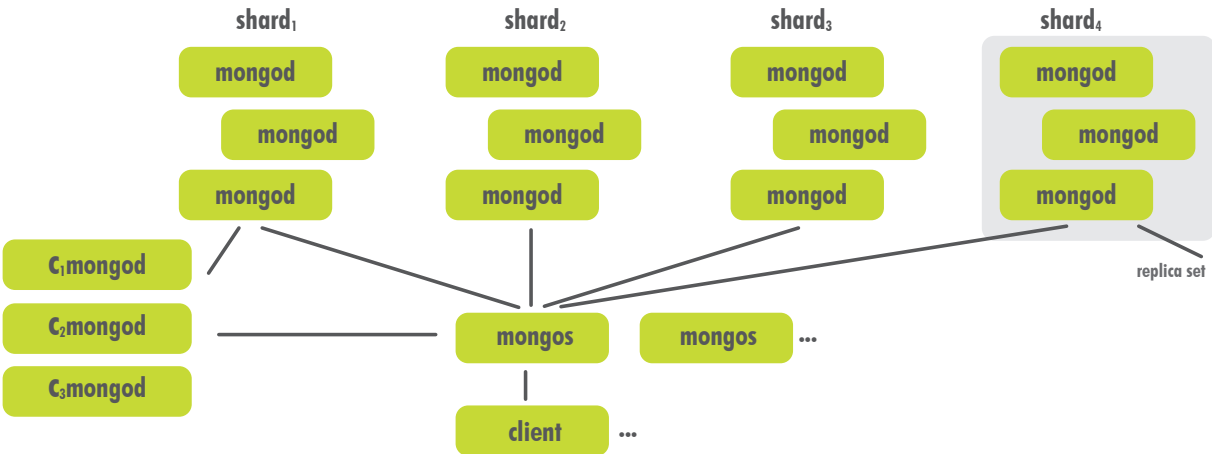
The tools traditionally used to scale RDBMS implementations have centered on packing more CPU’s and RAM into a server (vertical scaling), and complex SAN environments to manage large arrays of disks. These tools are often unavailable in the cloud, replaced by commodity hardware with very different performance characteristics.

**INTRODUCING MONGODB**

MongoDB is the leading open-source NoSQL data store, driving the market evolution. It was designed from the ground up to specifically address these new workloads and computing environments, completely changing how data is modeled, stored and accessed.

**Horizontal scalability**

MongoDB is horizontally scalable. Rather than buying bigger servers, MongoDB scales by adding additional servers. While Moore’s law is still intact — transistor counts still double every 24 months — improvements come in the form of more processors and cores rather than faster processors.



Built to handle large data sets, MongoDB's use of multiple servers means you have all the resources you need to add compute, memory and storage capacity. As your data set gets bigger, there is no need to upgrade to expensive high-end hardware. This also means you can incrementally adopt newer and faster compute platforms without throwing out the models you had before.

MongoDB easily supports high transaction rate applications because as more servers are added, transactions are distributed across the larger cluster of nodes, which linearly increases database capacity. With this model additional capacity can be added without reaching any limits.

### **Developer productivity**

MongoDB offers a data model and query API that is more agile and better suited to modern development stacks and methodologies than traditional data stores. Rich objects are stored in hierarchical documents rather than rows split across multiple tables. These expanded data models result in expanded documents, rather than new rows, tables and columns. As a result, transactions remain simple even as data models evolve. If, for example, ten new fields are added to a document, the query time to fetch the document does not increase.

Documents in MongoDB use a flexible schema and can change dynamically with the continual development of your application. There is no need to develop a rigid schema that requires transformations of data and management of schema migrations in production. If your application data changes, fields can be added to objects without reconfiguring your database.

### **Cloud ready**

MongoDB was designed to run on commodity hardware, virtualized infrastructures, and the cloud. Capacity can be added through additional virtual machines on demand, enabling an elastic, scalable database on whatever infrastructure is present. This means that cloud and hypervisor-based environments are just as suitable as dedicated hardware. Additional virtual servers can be used to compensate for the varying performance and capacity of individual server nodes.

There are no limits to where you can run your application. Your developers, QA, staging, and production environments can use the same code without worrying about sharing proprietary or expensive hardware platforms.

### **Growth Without Bound**

By scaling across multiple servers, MongoDB ensures that your application will grow and run without bound in cloud and virtualized environments. And because MongoDB's data model matches today's data requirements, your developers will be more productive than with competing solutions. With more than 100,000 downloads per month and industry leading support from 10gen, MongoDB is the perfect choice for your next application.



134 5th Ave.3rd Floor  
New York, NY 10011

100 Marine Parkway, Suite 175  
Redwood Shores, CA 94065

866.237.8815  
[www.10gen.com](http://www.10gen.com)