



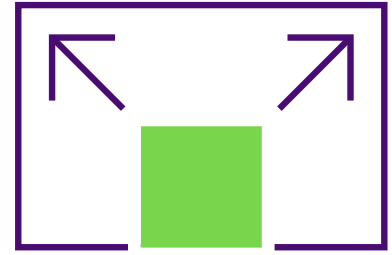
SUPER  
DATASCIENCE  
MAKING THE COMPLEX SIMPLE

# NOSQL ADVANTAGES



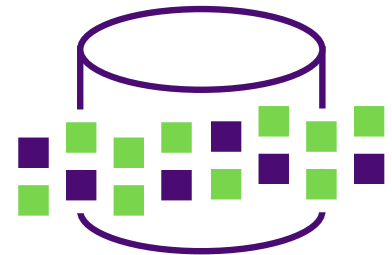
## 1. SCALABILITY

NoSQL databases, in general, are designed to scale horizontally (adding more servers to your cluster), while SQL usually only scales well vertically (increasing storage/processing power on existing servers).



## 2. BETTER HANDLE UNSTRUCTURED DATA

Because NoSQL databases don't require a predefined schema, databases don't need to have uniform structures. It's up to the developer/database designer to impose structures as desired.



## 3. GOOD TO BIG DATA

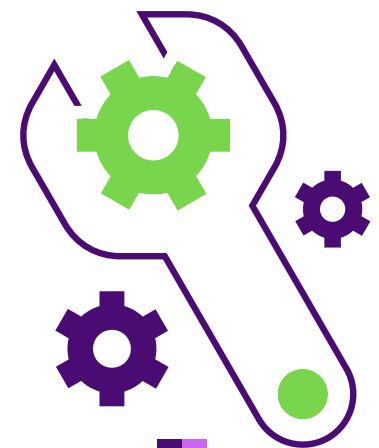
Due to scalability and ability to handle unstructured data, NoSQL databases are better suited for Big Data tasks. The volumes of “big data” that can be handled by NoSQL systems, outstrip what can be handled by the biggest RDBMS.



## 4. BREADTH OF FUNCTIONALITY

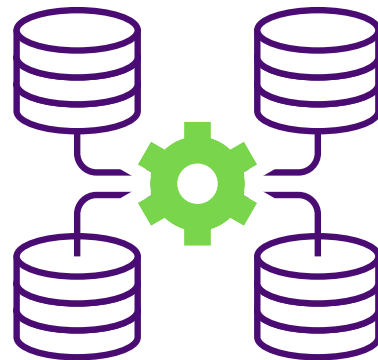
Most relational databases support the same features but in a slightly different way, so they are all similar.

NoSQL databases, in contrast, come in four core types: key-value, columnar, document, and triple stores. Within these types, you can find a database to suit your particular (and peculiar!) needs. With so much choice, you're bound to find a NoSQL database that will solve your application woes.



## 5. COST EFFICIENCY (SCALING AND MAINTENANCE)

In relation to the scalability benefit - horizontal scaling is more cost-effective than vertical scaling in general because there is an option to add low-cost servers to scale rather than having to upgrade to higher quality servers.



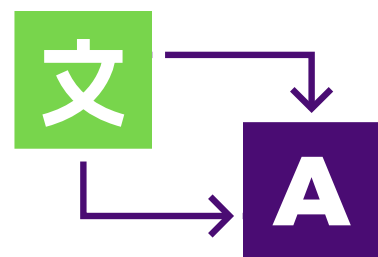
## 6. FLEXIBLE DATA MODELS

RDBMSs give colossal headaches when it comes to change management. The minor change must be carefully monitored, and may still involve some downtime or reduction in service levels. NoSQL does not have such restrictions on their data models, and even the more rigid NoSQL databases allow for relative flexibility like an addition of new columns with no major breakdowns.



## 7. NO SQL LANGUAGE

NoSQL databases support their own access languages that can interpret the data being stored. No need to be a SQL magician anymore to perform complex multi-table joins.



SUPER  
DATASCIENCE  
MAKING THE COMPLEX SIMPLE