

# Solution Brief

## NuoDB for the Telecommunications Industry

Exec  
Insight





## The Need for Information Technology Innovation

Mobile telecommunications providers are constantly striving to provide their subscribers the best service, and their ability to exploit information technology is a critical lever. Telecommunications has always depended heavily on information technology and a strong eco-system of innovative software vendors. Augmenting their own product development efforts has proven to be invaluable in helping the telecommunications providers meet these needs.

The on-going impacts of LTE (4G and beyond) means that what has been a good solution to date will not be adequate for tomorrow, so even the leaders have no time to rest on their laurels. And the depth and pace of the migration of consumer interactions to mobile for social media, entertainment, access to content and mobile commerce offer more and more opportunities for the best service providers to grasp rapidly expanding markets.

The combination of a changing technological environment to which they must adapt and new opportunities that are enabled by that change means that 'business as usual' is not an option. This leaves information technology providers, in-house and externally, with a Red Queen's race - they must run as fast as possible just to stay where they are, and twice as fast to make progress.



**A changing technological environment means that 'business as usual' is not an option.**



## The Cloud Challenge

The emergence of mature cloud platforms provides a new, more cost-effective way to deliver software services. It is an environment that provides a natural response to some of the challenges of the telecommunications industry. It matches the trend towards software-defined networks and away from the previously successful strategy of deployment of appliance-like solutions on dedicated, often specialist hardware in data centers.



**SaaS vendors adopt distributed architectures throughout the stack or struggle to serve mobile or geo-distributed consumers.**



### The decline of appliances

Although appliance and appliance-like solutions have the significant advantage of being easy to deploy, offering rapid time to value, they also impose significant constraints thereafter. Expansion (as workloads and data volumes grow) and upgrades of software can be difficult. Multi-data center operations are also likely to be complex, even if just for disaster recovery, never mind raising the bar to active-active or even active-active-active (more than two data centers). And yet elasticity of workload and distribution of data and workload are becoming essential requirements.

### The need for multi-data center operation

Multi-data center support is essential for resilience, even if this is achieved by active-passive fail-over. This has been a common approach for applications and has been largely adequate. However for databases that are not geo-distributed - and this includes all traditional SQL databases - multi-site operation is fraught with operational complexity and cost, as discussed above. As vendors migrate their offerings to a SaaS model, they must adopt distributed architectures all through the stack, and that includes the database, or they will struggle to offer an effective service to mobile and geo-distributed consumers.

### The impact of data residency requirements

One of the benefits of cloud deployment is the flexibility of location for compute and storage. But this brings with it another potential complexity, which is the need to manage data residency to comply with national and regional regulations. As providers seek to gain economies of scale and deliver unified cloud services into multiple geographies, they must adhere to sometimes-stringent residency regulations.

## The Implications for Software Developers

External demands placed on application developers by industry changes require architectural changes. The rising expectations of consumers lead to expanding requirements of the applications built. These factors, allied to the highly competitive nature of their business, place strong pressure on developers.

Factor	Response
<b>Peaky demand</b>	Provisioning systems for peak demand is expensive, especially when peaks can be many multiples of average. But mobile usage is often extremely peaky. Systems need to exploit cloud elasticity, which is a problem for (“OldSQL”) databases.
<b>Rising data and workloads</b>	Successful products incrementally add users and workload. The ability to smoothly provision increased capacity to meet these needs is essential to controlling costs.
<b>Competitive pressure</b>	Developers cannot control the service levels of their infrastructure, so they need to choose a platform that delivers agility, low touch administration and other benefits, such as rolling upgrades, to improve the service they can deliver to their customers.
<b>Software-defined network</b>	Appliances and appliance-like products raised the bar on time to value by reducing configuration and tuning time. But virtualization raises it still further, and is resulting in a move away from specialist-deployed hardware in customer data centers.
<b>Distributed service</b>	Telecom operators expect to deliver from multiple data centers, for resilience and lowered latency. Centralized, active-passive systems that operate disaster recovery from a second data center are not exploiting the opportunity to deliver service locally. They also add complexity and cost and a potential for system outages. Whereas NuoDB can offer continuous availability and active-active multi-data center operation.
<b>Multiple customers</b>	As software vendors add new customers, one of the potential benefits of cloud architecture is the option to deliver multi-tenancy systems - multiple customer-specific databases that operate on shared resources, managed centrally. This offers greater simplicity and operational cost saving.

## The NuoDB Offering

What distinguishes the NuoDB database from other databases is that it uniquely combines rich SQL language heritage and strong ACID compliance in a fully distributed, elastic-scale-out, continuously available database. NuoDB's founders did not throw out the ACID and SQL babies with the scale-up straightjacket bathwater. They started from scratch to build a cloud-native, scale-out, distributed database that retains transactional consistency and full SQL access. These characteristics were designed-in from the beginning.

NuoDB engineers were fortunate that they had the opportunity to start from a clean sheet and did not have to try to reverse engineer distributed capability into a database architected and honed over decades to run on a single machine or close-coupled cluster of machines. NuoDB operates as a distributed, in-memory database, in which some nodes are responsible for persisting the cache to a durable store and others for servicing client transactions. We call this a Durable Distributed Cache (DDC).

“

NuoDB uniquely combines rich SQL and strong ACID compliance in a fully distributed, elastic-scale-out, continuously available database.

”

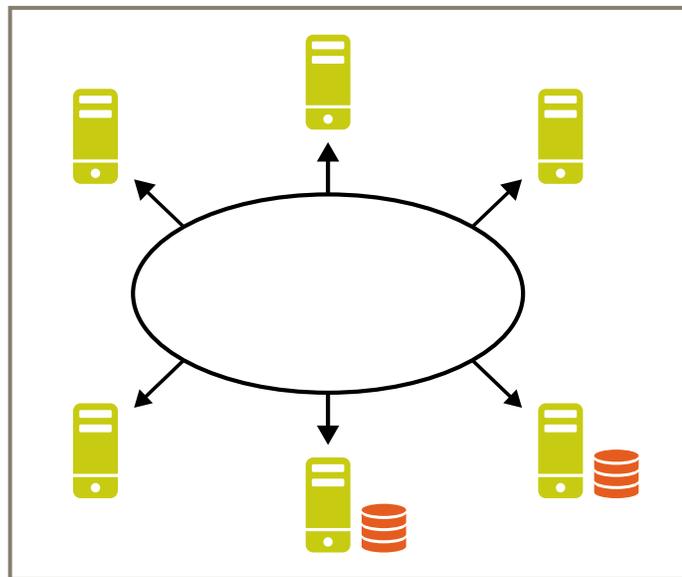


Figure 1: Durable Distributed Cache

## NuoDB Customers

Our customers tell us that they need to distribute their databases and access to those databases. They need to do this because their users are in different places and want low-latency access to data; they need to do this because their current databases are proving either very costly or impossible to scale-up as workload increases; they need to do this because they need to be “always up”, but their current architectures are complex, brittle and prohibitively expensive to maintain. And they need to do this in ways that do not lock them into expensive proprietary hardware, or appliances.

They need to do this in ways that can allow them to scale out and in elastically as demand varies, without permanently incurring the costs of peak capability when they only need it occasionally and temporarily. But



they are not prepared to abandon transactional consistency because they need to be sure every update is captured correctly. And they are unwilling to abandon their rich SQL heritage in skills, processes and tools to adopt immature programming models understood and usable by few, expensive, scarce resources.

Because NuoDB was architected from the beginning to address exactly these requirements, they are able to easily migrate existing applications to more virtualized and cloud infrastructures. And they are able to focus their new application development on the needs of their business, not on fathoming obscure new technologies.

### Case Study I

**Situation** Our customer is a successful supplier of software services that are white-labeled by network operators in North America, South America and Europe. Their main product, which has been very successful for its customers, is based on dedicated equipment and installed in operators' data centers.

**Issue** Network operators are now increasingly seeking to virtualize IT and so they favor cloud-friendly solutions.  
  
In addition, our customer was finding provision of disaster recovery capability and the necessary multi-site upgrades complex, challenging and brittle.

- Why NuoDB**
- Continuous Availability
  - Active-Active Distribution
  - Ease of Management

<b>Benefits</b>	Active-Active-Active	NuoDB's true multi-data-center operation delivers a better consumer experience as they roam across different geographies.
	Rolling Upgrades	The torturous upgrade process that carries real risk of outage is hugely simplified.
	Ease of Migration	NuoDB's rich SQL support enables straightforward and swift migration of existing SQL applications and tools to provide rapid time to value and lowered risk.



**NuoDB's true multi-data-center operation delivers a better consumer experience as they roam across different geographies.**



## Case Study II

**Situation** A major European telecommunications supplier has been piloting a mobile eCommerce application in small markets of up to one million subscribers. The application, based on a SQL database, has proved to be very successful, and the vendor is now looking to expand into larger markets.

**Issue** The architecture of their current implementation is limited in scalability without a big jump in cost and complexity. Even then it would still have a scaling ceiling, albeit a higher one. The data volumes are not huge, but the peak transaction rates in large markets easily exceed the 10,000 transactions per second limit of their current implementation.

**Why NuoDB**

- Scale-out performance
- SQL database

<b>Benefits</b>	Economic scaling capability	NuoDB empowers entry into more markets including more lucrative markets.
	Cloud offering	NuoDB's cloud-native, elastically scalable architecture provides attractive pricing and capability for virtualized environments.
	Ease of migration	The customer's team migrated their application to NuoDB unaided. A short tuning project elevated performance to beyond that achieved on the previous appliance implementation on comparable hardware.

“

NuoDB's cloud-native, elastically scalable architecture provides attractive pricing and capability for virtualized environments.

”

## How NuoDB works

Arthur C. Clarke, the distinguished science fiction author, once said: “Any sufficiently advanced technology is indistinguishable from magic”. And a SQL database that can maintain ACID-compliant transactions, active-active, across a geo-distributed network sounds a bit like magic. But in reality it is a radically innovative solution to a major problem of database technologies.

## NuoDB multi-layer architecture

The NuoDB architecture is split into three layers: an administrative tier, a transactional tier and a storage tier. The transactional and storage tiers are scaled separately and handle failure independently. Because of this, transactional throughput can be increased with no impact on where or how data is stored. Similarly, data can be stored in multiple locations with no effect on the application model.

This entirely new design, the Durable Distributed Cache (DDC), scales out and in dynamically, on commodity machines and VMs and has no single point of failure. Uniquely, it does all this while delivering full ACID transactional semantics.

### Transactional processing

In a DDC, Transaction Engines (TE) manage client transactions. For example, a TE needs the record for Customer #17; it determines that the data for Customer #17 is cached on a number of other servers. It gets the data from the most responsive server, as a remote memory fetch. But TEs can only share data they already have cached; if a request cannot be served from cache, a Storage Manager (SM) will serve the data. SMs don't support transactional connections to clients; they only manage the persistence of data and serving that up for the TEs.

All servers in the DDC architecture can request objects and supply objects. Any one TE has only a subset of all the objects in the database cached at any time, and can therefore only supply a subset of the database to other TEs. SMs, collectively, have all the objects and can supply any of them, but will be slower to supply objects that are not already in memory.

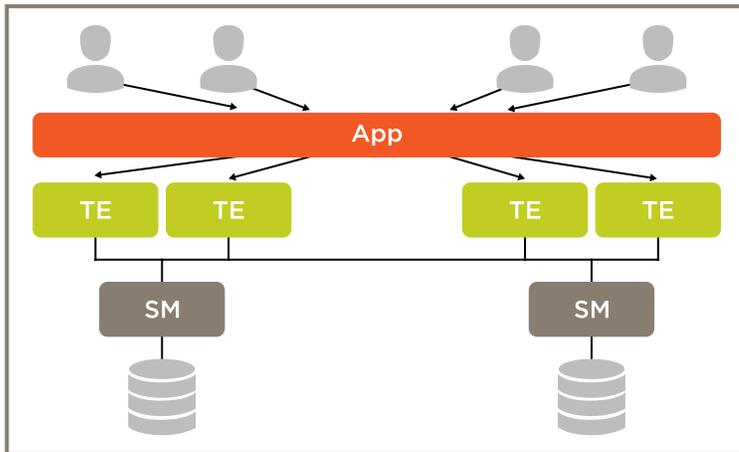


Figure 2: Multi-level architecture

### Resilience to Failure

Any data in a TE's cache is either already committed to the disk (that is, safe on one or more SMs), or is part of an uncommitted transaction that will simply fail at the application level if the object goes away. This means that the database is resilient to the loss of TEs. You can shut a TE down or just unplug it, and the system does not lose data. It will lose throughput capacity of course until a replacement TE is spun up, and any partial transactions on the TE will be reported back to their client applications as failed transactions. If the client restarts the transaction it will be assigned to a different TE to proceed to completion. This means the DDC architecture is resilient to the loss of TEs.

SMs scale out as needed, in the same way that TEs do. Database designers use NuoDB's Storage Groups to map sub-sets of the database to specific

“

The Durable Distributed Cache maintains ACID compliance, scales out and in on commodity machines, and has no single point of failure.

”

SMs. They may do this to satisfy residency requirements, to localize data with its main users or to enable scale out of data and I/O performance. Multiple SMs also mean the DDC architecture is resilient to the loss of individual or groups of SMs.

## Geo-distributed ACID transactions

Transactions, specifically ACID transactions, are an enormously simplifying abstraction that allows application programmers to build their applications with very clean, high-level and well-defined data guarantees. Application programs are vastly simpler when they can trust an ACID-compliant database to look after their data. The DDC architecture adopts MVCC (multi-version concurrency control). MVCC confers a number of well-known benefits, most notable of which is that it allows a database to manage read/write concurrency without centralized locks. In NuoDB, readers don't block writers, and writers don't block readers. Even more importantly, almost all inter-server communications are asynchronous, so network and server latencies are much less critical. A NuoDB instance can start TEs or SMs in a remote data center and connect it to the running database. Or it can start up some of the database servers in a private data center and the rest on a public cloud. Modern applications are distributed. Users of websites are spread across the globe. Mobile applications are geo-distributed by nature.

“  
Mobile applications need a unified database, running on a group of database servers in multiple locations.  
”

These applications are not well served by a single monolithic database server. Nor are they well served by clusters of sharded, separate databases that do not cope well with roaming users. They need a unified database, running on a group of database servers in multiple locations. That gives them scale-out performance, data center failover and the potential to manage issues of data privacy and sovereignty.

## Summary

The telecommunications industry offers a large and strongly growing market for specialist software and technology suppliers. Their customers, the network operators, are seeking to master the multi-dimensional challenges and opportunities of rapidly evolving regulatory, technological and customer pressures concurrently.

In this vibrant, even turbulent, environment, innovative and agile suppliers have a great opportunity to prosper, but to do so they need the right tools and processes. They already have, for the most part, a huge investment in SQL-based code and skills, but their OldSQL databases are ill-equipped for this new world. NoSQL databases have already staked their claim in this space, but migrating to these new platforms, with its accompanying loss of ACID transactions and inability to support anything but the most rudimentary SQL, brings with it huge cost and risk.

NuoDB offers a resolution of this dilemma; it provides a modern platform, designed from inception for the challenges of cloud deployment and geo-distributed database.

To learn more about the NuoDB geo-distributed database, our customers, and our solutions, visit our website, [www.nuodb.com](http://www.nuodb.com).

**215 First Street  
Cambridge, MA 02142  
+1 (617) 500-0001  
[www.nuodb.com](http://www.nuodb.com)**

TSB 06151

© 2015 NuoDB, Inc., all rights reserved.

The following are trademarks of NuoDB, Inc.: NuoDB, NewSQL Without Compromise, and Nuo.

