

# Key Features of Xpand

- [Scalable](#)
- [High-Volume, High Concurrency OLTP](#)
- [Automatic Data Distribution](#)
- [Distributed Query Execution](#)
- [Fault-Tolerant](#)
- [Flexible Deployment Options](#)
- [MySQL Compatible](#)
- [Easy to Migrate from MySQL](#)

Xpand is a full-featured RDBMS that combines a sophisticated distributed architecture with a simple SQL interface. It was specifically built for online transaction processing (OLTP) as well as for MySQL compatibility.

To better enable you to understand this document, some definitions are in order. Within Xpand, a **node** refers to an individual networked server while a **cluster** consists of three or more nodes that are configured to work together.

Some of the key features of Xpand are as follows:

## Scalable

Xpand uses shared-nothing architecture; the only architecture that is known to scale linearly as nodes are added. In a shared-nothing architecture, each node owns a portion of the data. Reads and writes are distributed to multiple nodes to reduce contention. Additionally, Xpand automatically distributes both data and query execution to scale.

Xpand further allows you to easily add ("Flex Up") and reduce ("Flex Down") capacity of your database to meet the changing and seasonal requirements of your application. To expand your cluster's capacity, use the [Flex Up](#) feature to add nodes to your cluster. Conversely, use the [Flex Down](#) feature of Xpand to scale back your configuration. Both operations automatically redistribute data within the cluster in the background while the database remains online and available.

## High-Volume, High Concurrency OLTP

Xpand was built specifically for large-scale, online transaction processing (OLTP). It ensures that transactions always maintain ACID compliance, even in a distributed environment. Xpand is a fully relational database built for high throughput workloads and can scale reads and writes by simply adding nodes. By distributing data and workload across different nodes and cores in a cluster, Xpand can achieve levels of parallelism that far exceed what is possible on a single instance database.

## Automatic Data Distribution

A key component of Xpand is the [Rebalancer](#) which runs continuously in the background to automatically manage the distribution of data for the cluster. It ensures that multiple copies (replicas) of your data are maintained across the cluster. If the cluster encounters an unexpected node failure and there are not sufficient replicas available, the Rebalancer automatically works to create more. Both data and load are distributed evenly across the cluster, even as data is being added or removed.

## Distributed Query Execution

Each node of a Xpand cluster is configured with the same version of the database engine, a map of all the system's data, and its own query compiler. Each node is capable of performing both reads and writes.

Queries are evaluated by Xpand's powerful [Query Optimizer](#), Sierra, to determine the optimal execution plan. The database then dissects each query into individual segments that are distributed to the nodes containing the relevant data. In essence, Xpand delivers the query to the related data and then amalgamates the results. All of this sophisticated distributed query planning and execution is leveraged by a simple SQL interface.

## Fault-Tolerant

Xpand was designed for fault tolerance and by default, can sustain a node or zone failure with no data loss. When a node or zone becomes unavailable, it is removed from service and the cluster is able to leverage copies of data on other nodes. The Xpand Rebalancer will automatically create additional replicas of that data, a process that happens transparently in the background with no user intervention and, as such, makes Xpand auto-healing. To learn more, see the article regarding Xpand's [Consistency, Fault Tolerance, and Availability](#). To learn more about deploying a cluster across multiple zones, see [Zones](#).

## Flexible Deployment Options

Xpand can be deployed wherever CentOS 7.4+ is available -- in clouds (AWS, Rackspace, Azure) or on commodity hardware of your choice. For more on recommended hardware platforms, see [Xpand Reference Server Configurations](#).

## MySQL Compatible

Xpand utilizes MySQL syntax and constructs for SQL, DML, DDL, triggers, and stored procedures. For current MySQL users, that means that your present environment is likely already compatible with Xpand. Although similar to MySQL, Xpand was actually built from the ground-up. See [MySQL Compatibility](#) for more information.

Xpand also supports the MySQL Replication protocol, including SBR and RBR. For more information, see [Configuring Replication](#).

Users are provided with a simple SQL interface for accessing data within Xpand. An application will see Xpand as a single instance, regardless of how many nodes are in the cluster or where the data is located. Xpand does not shard data and no modifications to the application are required to access the database. Each node within Xpand is able to service any reads or writes.

## Easy to Migrate from MySQL

Because Xpand speaks MySQL, migration utilizes both existing MySQL tools and proprietary Xpand products. A typical migration looks like this:

- Dump the MySQL database with `mysqldump`.
- Import the dump Xpand using `clustrix_import`.
- [Migrate users and permissions](#)
- Use replication to synchronize Xpand with the most recent changes made to the MySQL database.
- Cut over application servers to Xpand.

For detailed information regarding migration, see [Migrating to Xpand from MySQL](#).