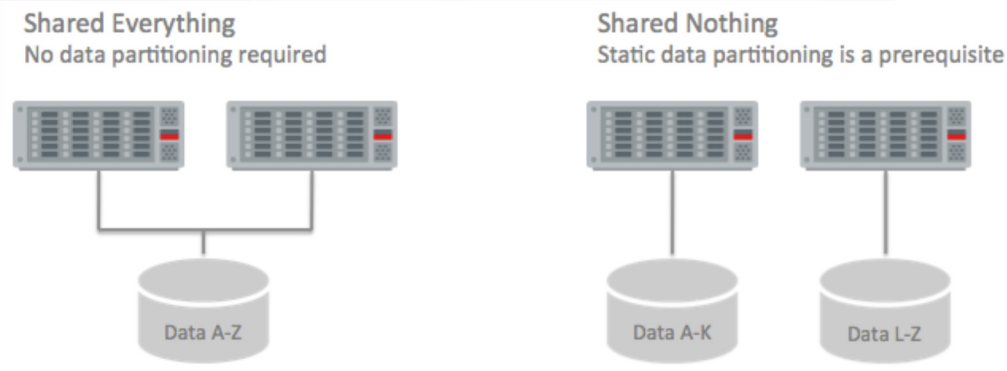

Figure 1: Shared everything versus shared nothing

In a **shared nothing system**, the system is physically divided into individual parallel processing units. Each processing unit has its own processing power (CPU cores) and its own storage component; its CPU cores are solely responsible for its individual data set on its own storage. The only way to access a specific piece of data is to use the processing unit that owns this subset of data. Such systems are also commonly known as Massively Parallel Processing (MPP) systems. Data partitioning is a fundamental prerequisite for these systems. In order to achieve a good workload distribution shared nothing systems have to use a hash algorithm to statically partition data evenly across all available processing units. The data partitioning strategy that controls the data placement has to be decided upon initial creation of the system.

5.12.4

As a result, shared nothing systems introduce mandatory, fixed minimal parallelism in their systems in order to perform operations that involve table scans; the fixed parallelism completely relies on the fixed static data partitioning at database or object creation time: the number of parallel processing units determines the minimal degree of parallelism to access all partitions of the data. Most non-Oracle data warehouse systems are shared nothing systems.

Oracle Database relies on a **shared everything architecture**. This architecture does not require any pre-defined data partitioning to enable parallelism; all of the data is accessible from all processing units without limitations; the degree of parallelism for an operation is decoupled from the actual data storage. However, by using Oracle Partitioning, Oracle Database can operate on the same processing paradigm, offering the exact same parallel processing capabilities as a shared nothing system. It is worth noting that it does so without the restrictions of the fixed parallel access encompassed in the data layout. Consequently, Oracle can parallelize almost all operations in various ways and degrees, independent of the underlying data layout, in addition to the parallel capabilities of a shared nothing system. By using a shared everything architecture Oracle allows flexible parallel execution and high concurrency without overloading the system, using a superset of parallel execution capabilities over shared nothing vendors.