

WHITE PAPER

# VERSANT OBJECT DATABASE PAIRED WITH THE KOVE XPD2 STORAGE SYSTEM

By Matthew Barker, Versant Corporation

## Versant

- » Direct transparent persistence of your object model for C++ and Java applications
- » No-SQL technology eliminating OR-mapping and joins while providing world class performance and scalability
- » Ideal for complex data, graph oriented data, highly distributed, or Big Data applications
- » Green technology requires minimal CPU processing power and low TCO

## Kove

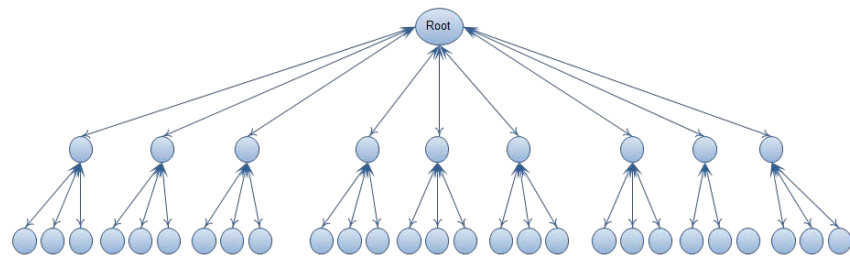
- » Kove XPD2 is the world's fastest and highest density memory disk
- » DRAM based storage solution with exceptional price to performance ratios, functionality, and scalability
- » Provides continuous, predictable, sustained I/O in a "green" technology
- » True Enterprise solution since the XPD2 can operate as a standard mesh sharable device

## ABSTRACT

The Versant Object Database is well known for its outstanding performance, particularly for data ingestion of complex models including highly connected graph data. In this Versant white paper, we compare the performance of the Versant Object Database using a traditional RAID storage system versus VOD utilizing the Kove Data Storage system for creating trees of interconnected objects.

### BENCHMARK OVERVIEW

For our benchmark test, we create N-ary balanced trees (constant fan-out). Each node is a simple C++ object with variable sized payload that maintains bi-directional relationships between child and parent nodes. Each node maintains an array of persistent links to its children and a single link back to its parent node.



**Balanced N-ary Tree**

Our benchmark runs parallel creator processes so we can compare execution times on a traditional RAID system and the Kove XPD2 storage solution as we increase the number of processes.

## BENCHMARK COMPARISON PLATFORM SPECIFICATIONS

### Traditional RAID Storage Based Solution:

- RAID 0 on PERC5 Dell Raid Controller
- Red Hat Linux 6.0
- Versant Object Database 8.0.2
- Dell PowerEdge 2950 Server

### Kove Storage Based Solution

- Kove XPD2 1 TB Storage Solution
- Red Hat Linux 6.0
- Versant Object Database 8.0.2
- Dell R815 Server

## BENCHMARK DESCRIPTION

The benchmark C++ application creates trees with a specified height, order (fan-out), and node size. The benchmark compares running simultaneous instances of the application all connected to the same Versant database. Our results include the use of 1, 2, and 4 simultaneous tree creator applications running against the Versant database using either the RAID or the Kove storage system.

### Medium Transaction Test

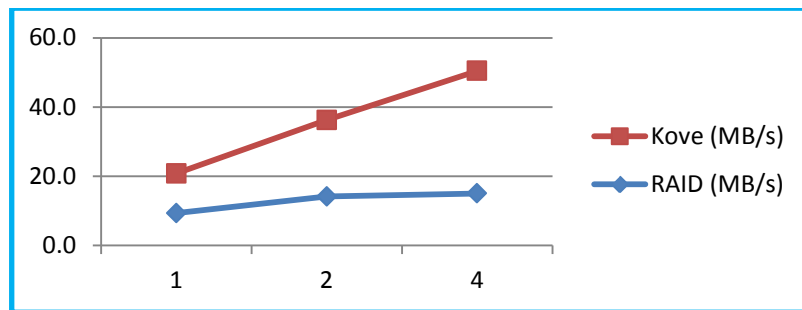
For our small transaction, each creator creates 4 trees with a height of 5, fan-out of 9, and node size of 1000. Each tree contains approximately 66K nodes for a transaction size of 6.6 MB's and a total of 266 MB's for creating the 4 trees.

### Large Transaction Test

Our large transaction test, each creator creates one tree with a height of 5, fan-out of 12, and node size of 100. The resulting tree contains approximately 271K nodes for a transaction and total size of 27 MB.

## BENCHMARK COMPARISON RESULTS

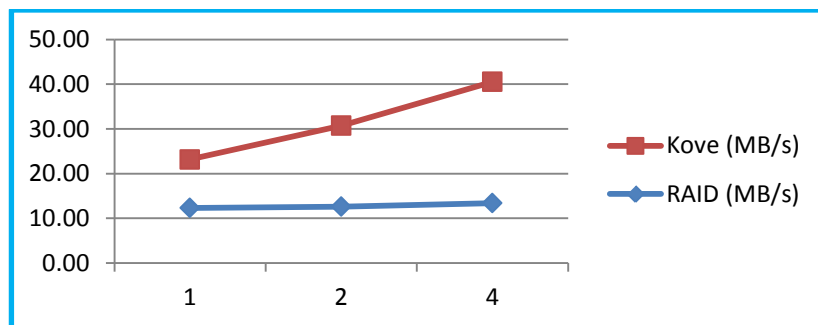
### Medium Transaction with 1, 2, and 4 Processes



### Interpretation of Results

The Kove system not only provides superior performance, but superior scalability as well. With a single process, the Kove system outperforms the RAID system by almost 2X. But as we add processes the RAID performance flattens out while the Kove system provides increased throughput as we increase the number of processes. These results also highlight the superior scalability of the Versant Object Database.

### Large Transaction with 1, 2, and 4 Processes



### Interpretation of Results

For larger transactions, the Kove system once again provides superior performance but the scalability advantage is even more prominent. The RAID system throughput is completely flat as we add processes while the Kove system continues to provide linear throughput as we add more processes connected to the same database.

---

## CONCLUSION

The Versant Object Database is the logical choice for applications with complex data persistence needs such as graph oriented data while Kove storage solutions provide record breaking storage performance.

### **Versant provides:**

- superior performance and scalability
- direct persistence of your Java or C++ object model
- no OR mapping and no joins resulting in fast, agile development and increased performance for complex data needs such as persistence of graph oriented data

### **Kove provides:**

- fastest available enterprise level storage operating at near memory speeds
- consistent performance across all I/O patterns regardless of reads or writes, small or large blocks, sequential or random, burst or sustained for any duration, at any occupancy.
- scalability, robustness, ease of use, and flexibility of direct- or mesh-connect storage

Combining Versant's superior persistent object technology with Kove's world record breaking storage solution delivers a persistent solution unmatched in complex data applications.

For more information go to [www.versant.com](http://www.versant.com) and [www.kove.com](http://www.kove.com)

### **About the Author**

**Matthew Barker** is the Director of Professional Services at Versant and has over a decade of technical experience with the Versant Object Database. You can reach Matthew at [mbarker@versant.com](mailto:mbarker@versant.com)