The first test was to measure the impact of caching. We made 1000 sequential buy/search requests for the same product, in 2 scenarios. One with caching, and one without caching. The table below reports the average time per request in each scenario.

	With caching (seconds)	Without caching (seconds)
Search	0.005859407901763916	0.02404384160041809
Buy	0.06658519172668458	0.05468181276321411

As we can see, searches become almost 5 times faster with caching. Also, buys become slower with caching as expected, due to overheads of maintaining cache consistency.

The second test we did was to measure the difference in latency of a request served by a cache hit and a cache miss. For this, we did the following 3 operations in sequence, 1000 times.

1. Lookup
2. Lookup
3. Buy

All operations are for the same product. The first lookup would be a cache miss, the second one would be a hit, and the buy would invalidate the cache causing the next lookup to be a miss again. We report in the table below the average time taken to serve a lookup request by a cache miss and a cache hit.

Lookup served by cache miss	0.02449019742012024 seconds
Lookup served by cache hit	0.011419462442398071 seconds

As expected, a request served by a cache hit is more than 2 times faster than one served by a cache miss.