



EXECUTIVE BIO

INNOVATING@SUN

Sun Storage J4000 – Part 2

Announcer You're listening to the Sun Microsystems Podcast Network. Welcome to another edition of Innovating@Sun with your host Hal Stern. Today's topic: The J4000 product line, part two. And now, here's Hal Stern.

HAL STERN:

Hello and welcome to another edition of Innovating@Sun. I'm your host Hal Stern, Vice President of Global Systems Engineering, and I'm joined again by Bill Moore who's the Chief Engineer of our Storage Products Group. And rather than talking about disks this time, we're going to go up the stack a little bit and talk about the software that drives the storage revolution and some other revolutionary announcements we've made, we think, regarding Flash memory. So, Bill, welcome back for part two of talking about "just a bunch of disks," just a bunch of software and just a bunch of Flash memory.

BILL:

All right. Thanks, Hal. Good to be back.

HAL:

And we're going to talk about an announcement that Sun did a few weeks ago clearly indicating that like most other vendors in the industry, we're going to be using Flash memory as another layer between main memory and disk and eventually tape. And Flash has a variety of really great features. You know, for example, a completely different power curve than the spinning media, but it also has a negative that's sort of a dark side to it, which is, it effectively wears out. So you're going to have other kinds of failures, I think, that we haven't really thought of that we have to make transparent to people. I don't want to have to go tell my customers, okay, it's great if you go use Flash memory, but make sure your applications go check to make sure that the data that they write gets read back correct. I mean, we've not been able to get people to go check the error on write system calls in 20 years. I don't think we're going to get application developers to go change now that we're changing the storage hierarchy.

BILL:

That's right and that's, again, what ZFS does on behalf of the applications. We actually have a 256 bit check sum on every single block of data that we pull in from the storage subsystem and, of course, that we write out to the storage subsystem. So in the case of Flash, the other important thing, I think, continuing on the thing you were mentioning about the reliability and also the performance is economically, it also occupies a very interesting space in the storage hierarchy; that is, it's a factor of four to five cheaper than DRAM but conversely, it's actually a lot more expensive per gigabyte than spinning disks. On the other hand, performance-wise, it's not as fast as DRAM, yet it's orders of magnitude faster than disk. So Flash, because it occupies both this performance niche and this economic niche between DRAM and disk means that it's actually a very interesting technology, and you see this in not only Sun picking this up, but everyone else. Where the big difference, though, comes is how we utilize Flash in our storage products because if you read the announcements from EMC, NetApp, pick your favorite storage vendor, what they're doing is they're introducing Flash as just another tier of storage, Tier Zero they often call it. It has to be managed. It has to be provisioned. It has to be utilized according to rules set up by the system administrator. And what we're doing with ZFS is we're taking advantage of both the economic and performance aspects of Flash, but also we're taking advantage of, as you put it, the darker side of the reliability characteristics of Flash in the way we utilize them. So in ZFS, in Solaris, there are two main ways in which Flash is utilized by the system. One is to accelerate writes. So whether you're operating a storage server that brings in data over the network and expects it to be synchronously written to disk or whether you have an application running on the box like your databases that expects that when the write

system call returns, it's actually on stable storage. In either one of those cases, the latency of that operation to complete is what drives the overall performance of the system. And in that case, rather than waiting the five to eight milliseconds it takes to get the data out to disk, we can – ZFS automatically will retarget those exact writes to Flash temporarily so that we can respond in a much, much lower latency to the application and it can get on with its life. And in the background, we actually migrate that data out to disk. So what winds up happening is that the application sees much higher throughput and a much lower latency than it otherwise would. That's on the write side. On the read side, we actually take advantage of the read performance of Flash, which is very, very good, 50 microseconds is a typical speed for an enterprise class Flash device. And what we'll do is we'll – ZFS through its adaptive caching mechanism will actually look at what data the application is using and decide, based on actual observed usage patterns, what we would like to hold onto in memory and what we can safely evict because the application most likely won't need it again. Of course, your application's working set probably doesn't fit in memory. If it did, well, you're in a very happy spot. But if not, again, what we're able to do with ZFS is take the data that we wish we could hold onto but just don't have room for, migrate that out to Flash which, again, is very, very scalable because it's out on your storage fabric acting as a very high-performance block device. And you're able to keep a much larger amount of your working set in a very low latency relationship to the processor and, thus, your application. So we're able to use ZFS to take advantage of these characteristics of Flash, the performance ones, and make a much faster system out of it. Now, as you pointed out, the dark side of Flash is that, yeah, it does have wear characteristics that have to be taken into account. So, again, in that case, that's where the checking summing of ZFS comes in, in that we're able to with, what is it, 77 nines of certainty, be able to tell whether the data we actually got back from the storage, whether it's Flash or disk, is actually the correct data. So once again, with Flash, in both the cases, the write acceleration and in the read acceleration, what we are able to do is if the data doesn't come off of those devices or if the write fails or the read fails, that's great. We can do what we've always done, which is write them to disk and read them from disk, you know, in multiple locations, obeying whatever rate policy we're supposed to and get the reliability that way. So the Flash doesn't act as something that has to be 100 percent reliable all the time. It's merely a performance accelerator that we can utilize, assuming that it's operating correctly. If it stops operating correctly, well, no skin off our nose. You're just back to the situation you were before Flash, which is –

HAL:

It looks like a bad block. I mean, so what, right? We've known how to solve that problem for a long time. So, Bill, one of the things that strikes me as you're talking about being able to incorporate things like Flash in a very transparent way now into the storage hierarchy is we're probably facing another set of disruptions in space, time trade-offs. I think that's one of the things you probably learned in the computer science curriculum is you always have to make space-time trade off sail. Either it doesn't fit in memory or you have to go solve for realtime so you're going to have to go make space optimizations, and you're always looking at the trade-offs essentially of what your constrained variables are. Now that we begin talking about massive amounts of data and as we increase our space, not giving up on the reliability side, as we add new tiers to storage, not having to make some of the same compromises on the time side or on the latency side, I get the feeling that we're just starting to scratch the surface now of what sorts of storage paradigms when I come up with – I think Thumper, the X4500, is one particular point in time. I think the first system we're going to see come out with Flash memory as part of the file system hierarchy will be another point in time. Where's the playground for this stuff? How are we going to go drive the innovation here?

BILL:

Yeah. So as I think you may have mentioned earlier on the show, or as certainly been mentioned heavily in the press by Sun over the last couple months is the open storage effort that we have underway, which is we have all of the building blocks in terms of software available for no cost for developers to download and try out, download the source code, modify it, try new features. And that's the huge advantage we have with the open storage stack. In addition, the expression of this open storage software on our hardware for which we can take, you know, special advantage of some of the features, I think gives Sun an entrance into a whole new potential line of products that you haven't really seen us focus on before. That is, we can take our industry-standard hardware which was designed actually in conjunction with the open storage software. That's another thing that I've been doing for the last couple years is working with Andy

Bechtolsheim on the design of these storage servers. And so what you'll see is you'll see both software that anybody can develop, and we have a huge community around this out on the opensolaris.org website, the ZFS one in particular is the most active community out in Open Solaris. And we're able to take all of that enthusiasm, all of that momentum and all of those increased number of features and co-design hardware to go with those that, you know, is designed with those features in mind. Sun's big strength, I think, is to act as a systems company controlling both sides of the equation, not just hardware, but software as well and designing everything top to bottom as an integrated stack.

HAL:

And I think that you mentioned early on about the rise of the general purpose system and really leading to some disruptions to market. I would argue that we've probably seen that happen once already in the pure compute space where open source operating systems, particularly Linux, combined with open source work management tools, open source file system and large data management tools really gave rise to, I would say, an enormous disruption in how people think about high-performance compute as this sort of whole new market has been born of that. And even at one extreme, you could argue that Google was born of that and they're sort of the poster child for using white-box PCs with a variety of open source software on top of them, hitting at a very different cost point and, therefore, a very different scale point than would have been possible a generation ago. Do you think we're going to see the same thing now with storage? I'll ask you the final question. Look out a little bit here. Is this disruption of the general-purpose hardware with this influx of creativity and influx of innovation through the open source community going to really reshape the kinds of storage systems that we build and, therefore, reshape the cost point of them for our markets?

BILL:

Yeah. Not only will it reshape the cost point. It will also reshape the capability point because the integration of all of these components, and as you mentioned, the amount of creativity that can be brought to bear on an open solution like what we have at Sun, is much greater than you would ever get with a closed, proprietary solution like you see in the market today. So I believe that it's not just something that might happen. I believe it's an inevitability based on the same exact driving factors that you just mentioned before on the compute side. All those same things, the open source software, the general purpose hardware catching up with proprietary hardware, all of that is coming to bear, I think, in exactly the same way on the storage side as has happened in the compute side over the last ten years.

HAL:

Okay. Bill, thank you for a great conversation about "just a bunch of disks," just a bunch of software and some of the disruptions we're likely to see as we scale up the problems of data management. You've been listening to another edition of Innovating@Sun and I'm your host Hal Stern.

Announcer You've been listening to Innovating@Sun. Join us next time for the latest in innovation from Sun Microsystems only on the Sun Microsystems Podcast Network.

[End of audio]