

How does the HP (Tandem) Non stop compare with Linux clusters?

Asked 16 years, 1 month ago Modified 5 years, 11 months ago

Viewed 5k times



4

HP NonStop systems (previously known as "Tandem") are known for their high availability and reliability, and higher price.



How do Linux or Unix based clusters compare with them, in these respects and others?



fault-tolerance

hp-nonstop

tandem

Share

Improve this question

Follow

edited Nov 10, 2010 at 14:34



Kelly S. French

12.3k ● 10 ● 64 ● 93

asked Nov 7, 2008 at 8:23



Abhishek Yadav

5,241 ● 4 ● 22 ● 10

1 Answer

Sorted by:

Highest score (default)



On a fault-tolerant machine the fault tolerance is handled directly in hardware and transparent to the application.

15

Programming a cluster requires you to explicitly handle the fault tolerance in the application.



In practice, a clustered application architecture is much more complex to build and error prone than an application built for a fault-tolerant platform such as NonStop. This means that there is a far greater scope for unreliability driven by application bugs, as the London Stock Exchange [found out the hard way](#). They had an incumbent Tandem-based system, which was quite a common architecture for stock exchange trading applications. Their new CEO had the bright idea that Microsoft was the way forward and had a big-5 consultancy build a .Net system based on a cluster of 120 servers.

The problem with clustered applications is that the failures can be correlated. If an application or configuration bug exists in the system it will typically be replicated on all of the nodes. This means that you can get a single situation or event that can take out the whole cluster. The additional complexity of clustered applications makes them more error-prone to develop and deploy, which raises the odds of this happening. A clustered system built on (for example) Linux and J2EE is vulnerable to the same types of failure modes.

IMHO this is a major advantage of older-style mainframe architectures. Several vendors (IBM, HP, DEC and probably several others I can't think of) made fault tolerant systems. The underlying programming model for

this type of system is somewhat simpler than a clustered n-tier application server. This means that there is comparatively little to go wrong and for a given amount of effort you can achieve a more reliable system. A surprising number of older architectures are still alive and well and living quite comfortably in their market niches. IBM still sell plenty of Z and I series machines; Unisys still makes the A Series and 2200 series; VMS and NonStop are still alive within HP. The sales of these systems are not all to existing clients - for example a Commercial Underwriting system (GENIUS) runs on the I Series and is still a market leader in this niche with new rollouts going on as I write this. The application has survived two attempts to rewrite it (1 in in Java and 1 in .Net) that I am aware of and the 'Old School' platform doesn't really seem to be cramping its style.

I wouldn't go shorting any screen-scraper vendors just yet ...

Gray & Reuter's [Transaction Processing: Concepts and Techniques](#) is somewhat dry and academic, but has a good treatment of fault-tolerant systems architecture. One of the authors was a key player in the design of Tandem's systems.

Share Improve this answer

Follow

edited Jan 23, 2019 at 20:34



marc_s

753k ● 183 ● 1.4k ● 1.5k

answered Nov 7, 2008 at 8:31



ConcernedOfTunbridge
Wells

66.5k ● 15 ● 148 ● 198

-
- 2 Was this the same London Stock Exchange project Microsoft boasted about a year ago or so in their advertisements?
– [Vinko Vrsalovic](#) Nov 7, 2008 at 8:39
-
- 1 Yes, the very one. At the time (mid 2005) I applied for a DW architect job at LSE and had this project described to me. I recall thinking 'Who the %!*&% signed that off?'
– [ConcernedOfTunbridgeWells](#) Nov 7, 2008 at 8:47
-