Twitter Thread by Dan Rose



Dan Rose
@DanRose999

I was at Amzn in 2000 when the internet bubble popped. Capital markets dried up & we were burning $1B/yr. Our biggest expense was datacenter -> expensive Sun servers. We spent a year ripping out Sun & replacing with HP/Linux, which formed the foundation for AWS. The backstory:

My first week at Amzn in '99 I saw McNealy in the elevator on his way to Bezos' office. Sun Microsystems was one of the most valuable companies in the world at that time (peak market cap >$300B). In those days, buying Sun was like buying IBM: "nobody ever got fired for it"

Our motto was "get big fast." Site stability was critical - every second of downtime was lost sales - so we spent big $$ to keep the site up. Sun servers were the most reliable so all internet co's used them back then, even though Sun's proprietary stack was expensive & sticky.

In 2000, brand new Sun servers started appearing on eBay for 10 cents on the dollar as VC-backed start-ups went out of business (this was pre-AWS when you had to roll your own datacenter). Amzn could have negotiated a better deal with Sun, but Jeff chose a more radical approach.

Amazon's CTO was Rick Dalzell - ex-Walmart, hard-charging operator. He pivoted the entire eng org to replace Sun with HP/Linux. Linux kernel was released in '94, same year Jeff started Amzn. 6 years later we were betting the company on it, a novel and risky approach at the time.

Product development ground to a halt during the transition, we froze all new features for over a year. We had a huge backlog but nothing could ship until we completed the shift to Linux. I remember an all-hands where one of our eng VPs flashed an image of a snake swallowing a rat

This coincided with - and further contributed to - deceleration in revenue growth as we also had to raise prices to slow burn. It was a viscous cycle, and we were running out of time as we ran out of money. Amzn came within a few quarters of going bankrupt around this time.

But once we started the transition to Linux, there was no going back. All hands on deck refactoring our code base, replacing servers, preparing for the cutover. If it worked, infra costs would go down by 80%+. If it failed, the website would fall over and the company would die.

We finally completed the transition, just in time and without a hitch. It was a huge accomplishment for the entire engineering team. The site chugged on with no disruption. Capex was massively reduced overnight. And we suddenly had an infinitely scalable infrastructure.

Then something even more interesting happened. As a retailer we had always faced huge seasonality, with traffic and revenue surging every Nov/Dec. Jeff started to think - we have all this excess server capacity for 46 weeks/year, why not rent it out to other companies?

Around this same time, Jeff was also interested in decoupling internal dependencies so teams could build without being gated by other teams. The architectural changes required to enable this loosely coupled model became the API primitives for AWS.

These were foundational insights for AWS. I remember Jeff presenting at an all-hands, he framed the idea in the context of the electric grid. In 1900, a business had to build its own generator to open a shop. Why should a business in 2000 have to build its own datacenter?

Cloud infrastructure would have emerged eventually even w/out AWS (like electric vehicles w/out Tesla), but how much later and at what opportunity cost ? After the cost of starting a company was reduced dramatically by AWS, innovation exploded and the modern VC ecosystem was born

Amzn nearly died in 2000-2003. But without this crisis, it's unlikely the company would have made the hard decision to shift to a completely new architecture. And without that shift, AWS may never have happened. Never let a good crisis go to waste!

PS: Amzn recently spent years ripping out Oracle, something few have attempted. It takes muscle to do hard things, and muscle gets built by doing hard things. The best companies look at every challenge as an opportunity and engrave that mindset into their culture.