Twitter Thread by BobCasey



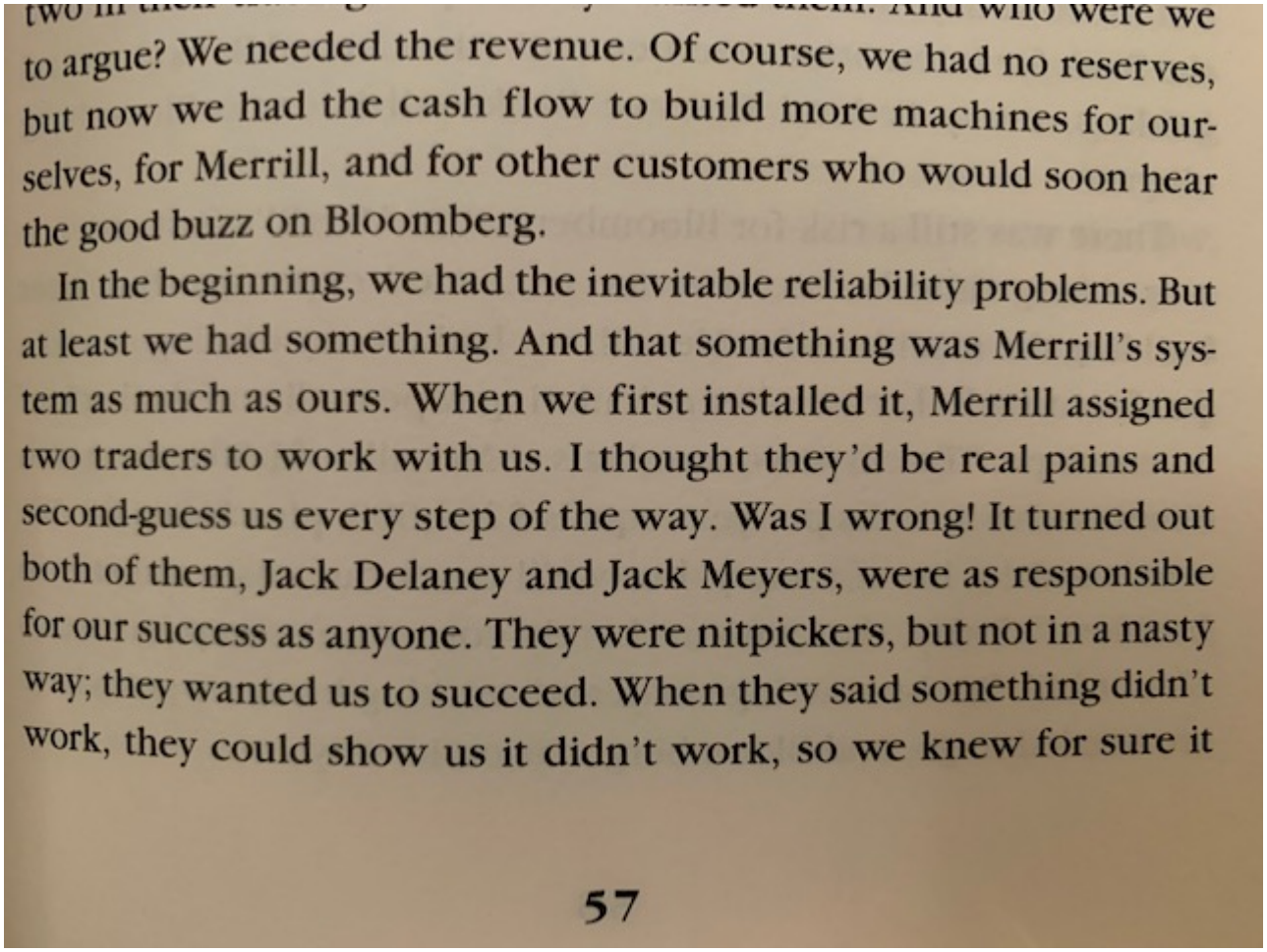
BobCasey
@BobCasey

1/ Read Bloomberg's autobiography (from '97) this weekend. Covers the first 15 years of Bloomberg LP & and how Mike built such a dominant franchise. Written before "lean startup" and modern software startup nomenclature, but lessons sound like something @chetanp would share...

2/ Bloomberg initially focused on a narrow use case: real time yield curve data for bond traders. Today, Bloomberg covers equity & debt markets around the world, offers financials, news, messaging, alternative data & more. But it started with a simple product & seemingly tiny TAM

3/ They "nailed" product/market fit for this use case because they deeply understood their customers. Not only did Mike Bloomberg and the first four employees all come from the industry, but they also built a tight feedback loop w/ their first "lighthouse" customer, Merrill Lynch

4/ When Bloomberg installed the first 22 terminals at Merrill, they embedded themselves with initial users. They visited daily. They had two users who provided constant feedback, helping to rapidly squash bugs and improve the product.



to argue? We needed the revenue. Of course, we had no reserves, but now we had the cash flow to build more machines for ourselves, for Merrill, and for other customers who would soon hear the good buzz on Bloomberg.

In the beginning, we had the inevitable reliability problems. But at least we had something. And that something was Merrill's system as much as ours. When we first installed it, Merrill assigned two traders to work with us. I thought they'd be real pains and second-guess us every step of the way. Was I wrong! It turned out both of them, Jack Delaney and Jack Meyers, were as responsible for our success as anyone. They were nitpickers, but not in a nasty way; they wanted us to succeed. When they said something didn't work, they could show us it didn't work, so we knew for sure it

57

5/ Bloomberg also offered professional services. In fact, their first $100,000 in revenue was from PS. The consulting work helped pay the bills, but more importantly helped the Bloomberg team map their customers' needs and to build relationships across the organization.

6/ As the product matured, one happy customer led to the next. The business grew largely via referral. Bloomberg shares how a sale to the Bank of England led to a sale to the Vatican led to a sale to the World Bank.

7/ Similar customers with identical use cases make for powerful references if a product relieves a pain point. The power of a narrow TAM helps not just with product market fit, but also with GTM.

8/ Bloomberg notoriously doesn't discount. Mike says: "To publish one price and then negotiate secretly with those you want to favor... that just encourages dissent, uncertainty, and unpleasantness." Many software companies struggle to maintain list price. Not Bloomberg.

9/ Mike reflects on the tension for a platform btwn offering 3rd party data vs producing content in house. Bloomberg News launched when this was an existential question for Bloomberg, as they began to compete w/ Dow Jones, which they also relied on for much of Bloomberg's content

10/ This tension is reminiscent of what we're seeing in 2020: Netflix's aggressive move into original content, Spotify's move into podcasts, and many other platform vs. content debates.

11/ Lots of lessons from Bloomberg for building a vertical SaaS business. Start with a specific use case. Iterate quickly w/ customer driven development. Build deep relationships w/ users and leverage happy customer referrals as your marketing machine. Strategically utilize PS

12/ Ensure your pricing model aligns with customer value and is fair across customers. Distribution & content both matter. If you can control both, you've got a powerful, deep moat. We're working hard on this @TegusHQ.

13/ A ton has happened at Bloomberg LP and in the broader financial world in the 23 years since the book published. I'd love to see an update on how Bloomberg LP has evolved, and am always looking for great stories about building successful software businesses. Suggestions?