What the PC Revolution Can Teach Us about the “Consumerization of IT”

The “consumerization of IT” is a name we’ve given to a new phenomenon. A couple of new edge devices have come on the scene – smartphones and tablets – and they are moving from being a consumer gadget to an element in enterprise IT. Employees are bringing them to work and insisting that IT support them, and IT departments have moved from ignoring the devices to trying to regulate them to accepting that they will have to support users who bring their own devices (“BYOD”).

But, as Yogi Berra said, in some ways it’s déjà vu all over again. In the late 1970’s and early ‘80’s a new edge device came on the scene. At first, the PC was a toy for prosumers and hobbyists; business IS departments – as we called them in those days – ignored the PC. They, after all, had to pay attention to more serious things, like how to get terminals and time-sharing to play with departmental computers as well as mainframes.

But, over time, employees began to bring the new devices into work, and then began to insist that IS support them. When PCs began to be connected into local-area networks to share files and printers, IS began to realize it had to support the new devices. The PC era had begun.

What happened in the ‘80’s and ‘90’s? How did the PC move from being a toy to being a key device in enterprise IT to being the killer of the UNIX workstation and server?

Most importantly, what can we learn about those days that can help us understand what will and won’t work, what will and won’t ensue, as smartphones, tablets, and other edge devices transform IT in the next few years?

Lesson #1: Platform Adoption is driven by Killer Apps

The spreadsheet was the original “killer app”: a software solution so compelling that users would buy a $2K piece of equipment – not supported by IT – in order to use it.

And so it was. When the IBM PC was delivered in 1983, it was something of a boat anchor until Lotus shipped 1-2-3, the first “business-grade” spreadsheet. 1-2-3 used the huge 640K memory space of the PC to keep the entire spreadsheet in memory, insuring decent calculation performance and crisp UI response. A star was born.

The word processor was a close second, and, over time, may well have been a more important innovation than the spreadsheet. The idea of being able to type documents on a computer, correct them ad infinitum, and only print them out when ready, transformed business communication and, later, when desktop publishing and laser printing came online, transformed typesetting and publishing.

For us in the consumerization era today, the early killer app is **email and calendaring anywhere/any time**. Consumers will buy a piece of equipment up to $2K in value (and, moreover, a monthly data plan for the equipment) in order to have that functionality. Other apps are important and fun – the PCs, after all, had Breakout and Adventure for fun, and CompuServe for "shopping" – but email and calendaring justify and drive purchase behavior. It is email and calendaring that drove Blackberry adoption a decade ago, and it is email and calendaring with rich clients that is driving iOS and Android adoption today. (Perhaps we should add "web access" to the current set of killer apps; certainly Blackberry users complained about the poor browser experience on that platform, and most of the apps in use today on "consumerization" devices essentially mediate web access to existing applications and sites.)

Lesson #2: Some (but not all) Killer Apps Drive IT Transformation

Looking back on the '80's, it's crystal clear what would follow the proliferation of PCs equipped for word processing and spreadsheeting: the desire, and then the need, to **share files and share file printing**. The first servers were file and print servers connected to workgroups over LANs (the invention and commercialization of Ethernet was of course a big help here). The killer apps drove a transformation of enterprise IT.

It may be obvious, but it's worth noting that not every killer app induces IT transformation. . What made the apps on the PC precipitate an architectural change was the requirements of the new uses they drove, in particular the possibility and then the requirement to **share files**.

Ten years ago the Blackberry emerged on the back of the same killer apps we see today: email and calendaring. But as long as the Blackberry was essentially a text terminal, there was only a weak imperative for IT transformation. Existing networks were good enough (fast enough and reliable enough) for message delivery. A new kind of server – the BES server -- was required for Blackberry operation, but adding BES servers didn't introduce any fundamentally new requirement into the architecture of the network. The Blackberry underwent a meteoric rise, but no architectural "tectonic shift" ensued.

What is driving IT transformation today is the heavier network payload of the newer devices and, more importantly, the new requirement that email and calendar be as reliable as non-mobile email and calendar. In the Blackberry "era" (if we can write it off as an era that has ended), a "best effort" approach to email and calendar synchronization was acceptable. To make delivery more reliable

- Single points of failure like the Blackberry regional servers have to go away.
- Sync and security conversations between devices and servers require lower-latency connections with fault tolerance built into the interaction.
- Devices need to use the fastest-available network, requiring handoff between next-generation cellular networks and Wi-Fi networks.

In short, today's edge devices drive IT transformations that were not necessary for Blackberries.

Lesson #3: PC Technologies Ultimately Disrupted Server Technologies

Ultimately, the PC revolution became "client-server computing", a profound change in enterprise IT architecture. An unexpected consequence of this IT transformation was that pre-PC server technologies were disrupted and replaced by the technologies used to build PCs.

It's worth dwelling on this a bit. At the dawn of the PC era, PCs and what we would today call "servers" were built with completely different technologies. VAXen were originally built with TTL, ECL, and MOSFET custom chip arrays, and, even when they began to migrate to microprocessors, used high-end custom chipsets.

By the mid-'90's clusters of "commodity" PC-based servers appeared in more and more serious applications. And ultimately, "scale out" servers based on PC technology destroyed other server technologies.

(Today, we are seeing the same dynamic with storage systems, where scale-out systems built on low-cost mechanical and, increasingly, solid-state technologies are displacing high-end technologies in both areas. The result will be a storage system architecture that looks a lot like scale-out servers did in the '90's.)

Ultimately, the PC, in addition to inaugurating an era of personal computing, led to a new architecture for business computing, client-server. For two decades, PCs, and servers ultimately built on PC technology, were central players in IT.

To some extent, this exact scenario is playing out again. ARM processors – the processors in many smartphones and tablets – are starting to migrate out of edge devices, and while this movement is in a very early form today, it may prove highly disruptive to existing server architectures. An HPCWire article on "The Processors of Petascale" from 4/10/12 ended with this observation:

> *So what does this mean for exascale? If history repeats itself, the first such systems will be powered by exotic processors, but eventually more commodity-based silicon will take over. Power consumption is the biggest constraint for large supercomputers now, so we may very well see other processor architectures like ARM come to the fore. And with the Chinese developing their own stable of HPC processors, the whole supercomputing game could change before the end of the decade.*[1]

The equivalent transformation of storage is well underway. Edge devices forced huge improvements in the manufacture and control of flash memory, improvements that are now migrating into enterprise storage systems in the form of solid-state disk (or "SSD").

Lesson #4: Apps are vital, but transient

The PC created a new industry, **shrink-wrapped software**, which sold what we today would call "apps" for price points of $20-$1000. Apps were required for the new functionality of PCs, and the health or illness of a PC platform depended on the availability of compelling and useful apps.

Both "compelling" and "useful" are important. Compared to the monochrome "green screen" apps of the time-sharing era, PC apps were attractive and almost "fun" to use. And of course they had to be useful. When the first web apps started to come out in the late '90's and early '00's, they were ugly and clunky compared to "native" apps, and users considered them second-class IT citizens.

[1] http://www.hpcwire.com/hpcwire/2012-04-10/the_processors_of_petascale.html

Yet over time, PC-based apps have diminished in importance. Today most PC functions are supported by satisfactory-to-excellent Web apps, and there are many connected functions that can't be done at all by a standalone app. The cause of this isn't hard to see: **app connectedness and TCO trumps app elegance**. If an app needs to be connected, eventually the client part of the app will atrophy in order to support lower TCO. And most apps today need to be connected. A disconnected PC in the '90's was an inconvenience; a disconnected PC in the '10's is almost a brick.

On mobile clients today, we are halfway through the life cycle of apps. Apps are important today because of 1) subtle (although very important) user experience benefits and 2) the need to persist local data against network fallibilities. Since connectivity is vital to most of today's pay apps, it's reasonable to expect the transition to technologies like HTML 5 to happen much more quickly on today's edge apps than happened on PCs. However rapid the change, the endpoint is the same: application and data logic migrate to the server side, spelling eventual doom for apps.

Lesson #5: Substantial Client-Side Hardware is Transient

The same logic applies to hardware capabilities on the client side. The need for compute and storage power on clients is driven by the lack of availability of same on the server side. If the server side were 100% available with zero latency, the client side could be a dumb terminal (as it was in the days of the mainframe) or even a virtual desktop (as it is becoming today).

It's logical to suppose that we will approach closer and closer to this availability and latency, so it's logical to expect that compute and storage capabilities will be less and less important on new edge devices.

Two hardware exceptions are **screen** and **keyboard**. Because of our physiologies, these need to be as big as possible on the client side (in fact, today's edge clients are insufficient in both screen and keyboard capability).

The reduced need for client-side compute and storage spelled the end of the workstation in the '80's. Compared to PCs, they were needlessly expensive. As network reliability, throughput, and latency improved, it was harder and harder to justify the expense of workstations.

An obvious "deer in the headlights" here is the laptop. The argument for a laptop is increasingly one of trading off cost and short battery life for screen real estate and a full-sized keyboard. Decent glass or wireless keyboards and larger tablet form factors should doom them.

Lesson #6: iOS vs. Android will not necessarily play out as Windows vs. Mac did

On the surface, the analogy between the PC situation in the '90's and the "new client" situation in the '10's is compelling: Apple on one side, with tightly combined and tightly controlled hardware, software, distribution, and design. On the other side, software openly available on a wide variety of hardware, in a loose, anarchic distribution and design "space." To hear Apple talk about it, the story in both cases is the same: shoddy goods drive out quality.

The real story is more complex. The original IBM PC and its clones had established a strong beachhead in businesses, while the Apple II, which had preceded them to market, remained a personal/home computer. Apple's attempts to position the Apple III and the Lisa as business computers had not been particularly successful. Apple had a reputation, at the dawn of the Windows/Mac wars, of not understanding or caring about the business market.

Mac pricing played a role in this. The PC clone marketplace drove prices down to a level consistently below the Mac's, and while individual business users may have wished to own Macs, corporate purchasers weren't having any.

But apps were also quite important, and maybe determinative. Even in those days, office "suites" had begun to have a network effect. Word and Word Perfect were the office choices for word processor. Lotus 1-2-3 was the default spreadsheet. Being able to exchange compatible spreadsheet and word processing files drove a "winner-take-all" market in office productivity apps which Microsoft won. And they won, not by monopolistic practices, but by virtue of having a product people liked and used. Certainly there was nothing for business office productivity on the Mac except for essentially the previous version of Microsoft Office.

There is a reason why new apps were preferentially deployed on Windows first. The Mac platform was harder to write software for, and had fewer outlets to sell it. Software developers figured at the time that Mac software was 3x as hard to write for and had 10% of the market size as PC software, implying that PC software had 30x the ROI of software for the Mac.

So a key question for today's "consumerization" ecosystem has to do with apps. Where will the app developers go? Where will the compelling apps be written?

So far, the loveliest apps have consistently appeared on the iOS platform, with Android apps lagging behind. Unlike the PC era, Apple has paid a lot of attention to nurturing an app developer community, and has provided tools, support, easy distribution, and automatic payment. Android apps are harder to develop, with less support, and with multiple hardware platforms and OS versions to take into account.

On the other hand, developers do not like the prescriptiveness of the iOS ecosystem. Only some apps are allowed, developers must ask permission for each change, and, perhaps the unkindest cut of all, Apple takes a big share of a developer's app income.

Lesson #7: "Consumerization of IT" is a Misnomer: It's Not Just About Gorgeous Devices

The "tip of the spear" in the '80's was the influx of "toy" PCs into business settings. But it would have been a mistake to call the movement in the '80's the "toy-ization of IT". What was going on was far more complex, as we have discussed:

- Applications like spreadsheet and word processing (which we discuss in more detail later) quickly led to use cases for sharing work files on file servers accessed over Ethernet networks.

- These use cases drove the deployment of LANs and file servers, changing office IT forever.
- The technologies used to make PCs began to tip over into servers, which in turn drove the development and massive deployment of "commodity server scale-out" architectures.
- And PCs themselves have become, more and more, dumb terminals in a new global server-oriented cloud architecture.

The phrase "consumerization of IT" captures something of what is happening now. Enterprise employees are demanding that their "consumer" devices like iPhones and iPads, be used as terminals for email, calendaring, and, soon, for other enterprise functions. Not just as terminals, but as first-class enterprise terminals, with full support from IT. What's compelling about these devices is their consumer-friendly user experience; it's actually kind of fun to use a "consumerization of IT" device.

But as we can see through the lens of the PC revolution, this characterization misses the point. What is happening is a multi-level transformation of enterprise IT driven by the usefulness of these devices:

- **Delivery of enterprise IT on a variety of devices.**
- **Enhanced networks – performance, security, availability – to support mobile and multi-client usage.**
- **Cloud backends with the multi-tenant on-demand architecture required to do minimum latency processing for the edge.**
- **Outsourcing of compute and storage from enterprise datacenters to service providers.**

One might as well describe this transformation as "cloudification of enterprise IT", or "Ubiquitous computing", or any number of other terms. Consumerization is only one aspect of the change. It's certainly as profound a transformation as the transition to client-server computing; it may be the most important transformation of business IT in our lives to date.

Conclusions

As Mark Twain said, "history doesn't repeat itself but it does rhyme." Analogies are a lovely but imperfect instrument for understanding. It is easy to match up correspondences from the PC era with trends in the current IT transformation and extrapolate what will happen based on what did occur.

However, there are a number of factors of similarity between the disruption caused by the invention of PCs and the disruption now ongoing with the "consumerization of IT". To reiterate:

- In the same way that the killer apps for PCs drove the transformation of enterprise computing to client-server, so the killer apps for "consumerization" devices are driving a new cloud infrastructure based on distributed server and storage capability and ubiquitous access
- In the same way that the technologies used for building PCs ultimately migrated over to the server side, so the technologies for building "consumerization" devices seem likely to change how server-side compute and storage is built.
- Although an "open" challenger defeated Apple for hegemony in the PC desktop market, it is not yet clear whether Android will prevail over Apple today. The game goes to the side with the best apps.

- Although apps are vital today, their use value will wither as the cloud/ubiquitous infrastructure improves, which is what happened to desktop apps (and indeed to desktop computers) over the course of the client-server transformation.
- Just as the advent of the PC was the beginning of client-server (and then really of Web) computing, so the consumerization disruption today is the beginning of a transformation to the “cloud-ubiquitous” architecture of the next wave of IT.