What killed Motorola? Not Google! It was Moto's dire software

The Register (http://www.theregister.co.uk/2012/11/29/rockman_on_motorola/) · by Simon Rockman · November 29, 2012

Feature The Motorola we used to know is dead. After it was split in two, Google bought phone maker Motorola Mobility in May this year, leaving the profitable equipment biz Motorola Solutions to live on.

Then the ad giant decided to lay off 20 per cent of the Moto Mobility workforce (http://www.theregister.co.uk/2012/08/13/google_motorola/). Although this was all relatively sudden, the fall of the company - an outfit that opened its doors nearly a century ago - can be traced back to the late 1990s.

The stated aim of the Googorola love-in was to foster a boutique manufacturer of devices to run Android, Google's mobile operating system. Google was also keen to get its hands on Moto's incredible library of intellectual property, which included many great innovations in radio technology, mechanical engineering, electronics and production engineering. Yet many of the people who made this happen are busy updating their profiles on LinkedIn. It seems that Google isn't looking to Motorola to be a source of future intellectual property.

(The top designers at Moto are not hard to spot, by the way - just look at the staff passes; get a patent awarded and it goes on your company record. Collect 50 and you get a gold badge with your name on it.)

How did Motorola grow to become one of the biggest companies in the world, and at one point hold a dominant market share? In a word: hardware.

And how did Motorola lose it, and end up with the major part of the company bought by Google? In a word: software.

Where it all started: The Galvin brothers' golden years

Motorola’s background is radio: it was founded in 1928 as Galvin Manufacturing Corporation by brothers Paul and Joseph Galvin. It was run by members of the family until 2004 when the board ousted Paul's grandson Chris. The name change came in 1947 when the company's car and two-way radios sold under the Motorola brand were well established.

Sales of equipment to the military were substantial: Motorola produced the first FM radios usable in the field, although the kit was the size of a rucksack. The company followed technology trends with television – including the world’s first rectangular TV tube - and, off the back of this, an early use of transistors. This led to chip design and manufacture.

Motorola's understanding of consumer and two-way radios in the 1960s meant it was well placed to work on emerging cellular communications. Motorola vice-president Marty Cooper demonstrated the first handheld phone in 1973 and it was a chunky beast. He said: "Battery lifetime was 20 minutes, but that wasn't really a big problem because you couldn't hold that phone up for that long." The first infrastructure to support calls from handhelds was sold in 1982, and in 1983 the DynaTAC phone was launched. The first GSM products were launched in 1991.

Martin Cooper with the DynaTAC prototype from 1973

Most of the heroes of the company worked in hardware. Ralph Pini joined as a bench engineer in 1976 and rose to CTO, leaving in 2004 to teach at the University of Illinois.

The move to digital consumer mobiles was hotly contested in the Motorola boardroom, but Pini made the right bet and is probably the person who created the value Google recently bought.

(http://www.ebay.co.uk/itm/1993-One2One-Motorola-m300-Original-Mercurys-first-Mobile-Phone-Spares-Untested-/261070608705?pt=LH_DefaultDomain_3&hash=item3cc9055941)

Old phones, classic look: One2One Motorola M300 (1993) and Motorola StarTAC (1996)

M300 source: Getmefixed.co.uk

But not every bet paid off. A couple of Motorola directors on a cruise thought it would be great if they could make calls while on holiday, and thus space communications biz Iridium was born. It has a network of 66 satellites giving global coverage to satellite phones and pagers.

The name comes from the element Iridium, which has an atomic weight of 77. This was the expected number of birds to form the satellite constellation, launched between 1997 and 1998, but in the end 11 fewer were needed. The other thing that was overestimated was the number of people who would want to buy the bulky and expensive sat phones.

Motorola spent $6bn on the project, but when the separate Iridium company went into Chapter 11 bankruptcy protection in 1999 it was bought by private investors for $25m. Those sugar daddies are believed to have a very close relationship with the FBI, and it's publicly stated that the biggest organisation that uses Iridium is the US Department of Defense.

While most consumers would rather have a cellular phone, others found significant advantages in a system that cannot be easily intercepted by the local authorities. As such, it has always been illegal to take an Iridium phone into Syria.

However the technical capability to build a satellite phone system is not to be sniffed at - and the ability to take a $6bn bath shows how big Motorola had grown in 60 years. The company has built infrastructure, the pioneering "professional social network" iDEN, two-way pagers, the first tri-band phone and the first GPRS phone. Talk to communications network engineers and you'll find they preferred the Motorola kit for radio performance. Moto's expertise was so much greater than just making phones. Its chip division Motorola SPS, later called Freescale, pushed Motorola beyond building phones out of other people's parts: its deep understanding of radio was the best in the world and there was an academic focus on research and development.

These deep-rooted hardware skills were barely recognised by the senior management. Things just worked. Tri-band and quad-band radios, something all the rivals bar the Danish company Dancall struggled with, helped the company to compete confidently and globally.

It was the engineering focus that also led mobile operator Three to make Motorola its launch partner for its 3G network. The handsets were big and clunky, but it was three years before Nokia had something competitive.

For the first 60 the company's brilliance in hardware and excellence in radio meant Motorola grew spectacularly. That superiority never faded, but as software started to become more important it wasn't enough.

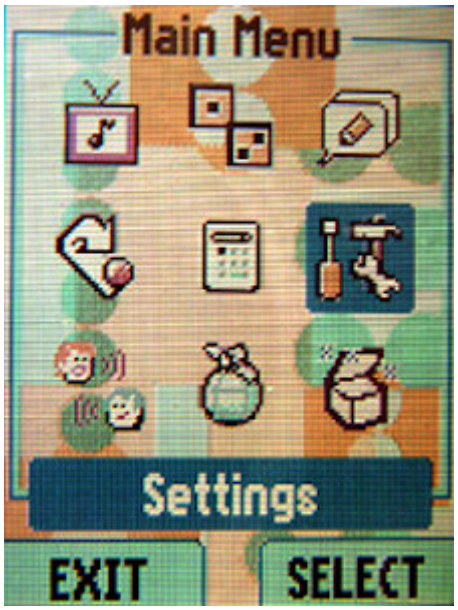
What completely derailed Motorola? Software

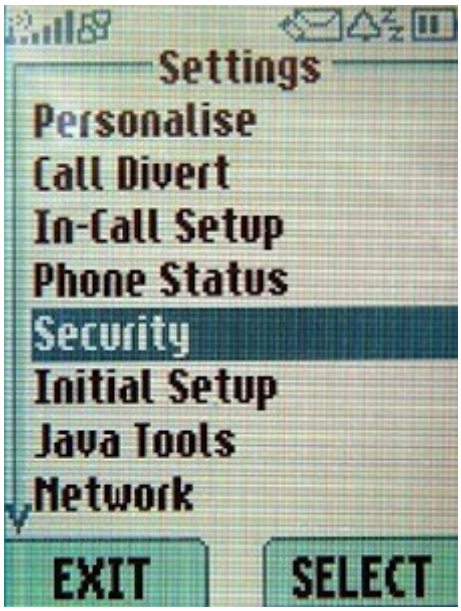
What Motorola failed to do was spot the turning tide in what people wanted; it became a time when, for instance, you bought computer games on the strength of who wrote them: if Minter, MacLean or Molyneux was on the tin, you stumped up the dosh. The result was none of Motorola's great talent wanted to work for the company. Why program for a dull Midwest company when you can go to the Bay area and be fêted?

As screens evolved beyond seven-segment displays, phones needed fluffy user interfaces. They needed more features – such as customisable ringtones, clocks, calendars, games, biorhythm calculators and other tat.

Christian Lindholm at Nokia saw this and started using psychologists to understand what people expected phones to do. Motorola developed software it called Personality, which allowed users to configure their primitive handsets the way they wanted. But the code struggled to keep up on the manufacturer's MicroTac phones so a plan was hatched: a new elegant platform, something like an operating system, would sit between the user and the firmware that did all the real work of controlling the radio and the UI. And so Platform 2000 was born, and sitting atop P2K was the Synergy user interface. This combination was to be the future for Motorola’s phones from analogue and US TDMA to CDMA and GSM - one platform to unite all devices.

To shortcut the development time, the new software wasn't written from scratch but based on the existing Personality code. It didn’t come soon enough.





Remember how phones used to look? The Synergy user interface in 2002

Phones scheduled to be on the new platform started to run late. Quick kludges were implemented to get gear shipped with the software. One phone would be shipped with new features banged in but written in a way that could not be absorbed back into the operating system's mainline source code. This prevented other phones from easily including the feature: there was no time to redo what had been hacked together for other hardware.

The software forked like crazy. Building a version of the OS for each new phone became a major undertaking. A colour-screen GSM phone with photos in its address book could not pass along this feature to other devices.

Bit hard playing games if you can only press one button at a time

At the same time, requirements soared: Moto's software engineers needed to add support for colour screens, cameras and Java. In 2001 there were still awkward limitations in the system, such as the inability to detect more than one button on the keypad being pressed at once – something games needed if the player wanted to move and fire at the same time.

More and more development resources were thrown at P2K with thousands of people working on it. The failure to manage the operating system programmers meant they churned out stuff like crazy, bits of source code were lost, and eventually the platform became less like physics - where you know what will happen given a set of inputs - and more like biology - where you inject a drug and see what happens. No one wanted to make changes fearing what might break.

The solution adopted by the product development teams was to avoid P2K altogether and commission outside device manufacturers to make handsets using their own operating systems that were similar to P2K but did what the product managers wanted. Competition meant that a large number of these external electronics companies were used, each with their own software base. All the user interfaces had to look like Synergy. Keeping all the phones working in the same way became a nightmare. A simple change, such as adding a field to an address book, would have to be added to 20 code repositories and had a 36-month lead time.

Source code changes became such an overhead there were huge company processes put in place to make sure that everyone really wanted a change before it was implemented. The software requirements conference call was broken into a number of features, with fifty people invited to the daily call of an hour and a half, except on Fridays.

Motorola tried to redress this with the purchase of TTPCom for $193m (http://www.theregister.co.uk/2006/06/07/motorola_buys_ttpcom/) in 2006. The prize here was Ajar: the not-quite-open-software platform aimed at low-end phones. Handled correctly this could have saved Motorola's low-end business.

Unfortunately, TTPCom gobble was the brainwave of Moto's chief technology officer. The people who actually made phones – the product team vice-presidents - were so battle-scared from dealing with Motorola's in-house software, they didn't trust Ajar and preferred to use the proprietary operating systems built by the outside contractors, the original device manufacturers (ODMs).

For example, an ODM was told to build a product using either its own feature-phone OS or an Ajar-Synergy combination - whichever one could be done on time and to budget and looked like Synergy. If the ODM used its proprietary system, it would get royalties on each phone. If it used Ajar, it would not.

It was never a fair fight, made worse by the decision to take Ajar - a smart flexible multi-threaded OS - and port Synergy to it, running the whole environment as a single thread, effectively an OS within an OS. This was powered by a 50MHz ARM7. The memory budget went through the roof, it crawled along, and Ajar's reputation was tarnished within Motorola.

(If I sound bitter it's because I worked on a user interface for Ajar that was - in my humble opinion - smart, elegant and Nokia-beating. In six months with eight people we'd built something better, slicker and nearer completion than hundreds of Motorola bods had in years. Indeed, Motorola did so badly because it used armies of mediocre or, at best, good programmers. I worked with a few brilliant ones who could guide the merely good. By having everyone in the same room and using agile methods we raced ahead.)

Suddenly it wasn't just eight of us in a room in Cambridge - 30 people flew in

Senior product staff were like children playing a game of playground football, everyone chasing for the ball, and for a short while my project was the ball. Suddenly it wasn't just eight of us in a room in Cambridge - 30 people flew in for a two-day meeting. The eight of us had to stop work to host them. The result of the meeting was a proposal for another 30 two-day meetings. Every single person had their own view and plan, all empire building, with very little concern for the user and even less for getting anything to ship.

Before we had any of those follow-up meetings the project was reclassified as an "experiment", too small to be production ready, and killed. Thanks to the trashing of Ajar, none of the product people wanted to use it. One phone shipped running the platform with the Synergy user interface.

Taking into account the cost of buying TTPCom and then making all the staff redundant (http://www.theregister.co.uk/2008/01/15/ttpcom_layoffs/), plus development and slippage costs, that one phone must have cost close to half a billion dollars to build. It wasn't particularly good.

P2K and Synergy lived from 1998 to 2008. It never worked properly. While armies of developers were fighting code and politics on Platform 2000, there were two other battlefronts: Linux and Motorola doing the hokey cokey with Symbian - the mobile operating system developed in a joint-venture between Psion, Motorola, Ericsson and Nokia.

In 1999 Motorola signed a deal with Psion to build a Symbian phone: Odin (http://www.theregister.co.uk/2000/10/26/motorola_takes_wraps_off_odin/). This would have been just like the Sony Ericsson P800 but years earlier. However Motorola had so many different product lines under development the incoming Moto president Mike Zafirovski cut back to the core and most projects were canned including Odin. The loss of the project also caused the company to lose most of its EPOC-Symbian expertise.

The Moto A920

Symbian got a second wind in 2001 when the team rebuilt in Florida with a deal funded by the UK mobile network Three (http://www.theregister.co.uk/2003/09/22/motorola_is_reluctant_white_knight/). This saw the development of the A920 and A925, big and clunky 3G handsets excellent for web surfing, except that Three had a walled-garden policy for software and the phone never had any legs. Still it meant that the excellent A1000 was built. Unfortunately the sales people didn't get the A1000 and they never managed to capitalise on the success of the Symbian Sony Ericsson devices. Three phones and that was it.

Goodbye, Symbian

A corporate decision was made to move to Windows Mobile. It was seen that Symbian was an Evil Nokia thing, so Motorola sold its shareholding in the venture and watched Nokia turn the partnership against Motorola. Eventually the Symbian gang fell apart, leaving Nokia to layoff its engineers this year and bury the OS in an unmarked grave (http://www.theregister.co.uk/2012/10/19/symbian_maintenance_mode/).

But before then, in 2005, Motorola snapped up the engineering assets – all the bits worth having (http://www.theregister.co.uk/2005/06/29/motorola_sendo/) – of high-end Symbian mobe maker Sendo in Birmingham. Nokia's Series 60 phone software was now the enemy, not Symbian, and so in a fantastically expensive deal with Sony Ericsson, Motorola bought UIQ Technology, which layered a graphical user interface over Symbian.

I was at Sony Ericsson at the time and we could not believe our luck; we'd been struggling to get operator acceptance for UIQ and the Sony Ericsson P1 phone, but with Motorola on board, how could we lose? It was like Christmas. This third Symbian resurrection yielded the mechanically fantastic Moto Z8 and its Z10 sibling. The Motorola front-end to UIQ even had a nod to P2K Synergy.

Then Motorola shut down Sendo and the joint-venture of UIQ collapsed.

It's impossible to overstate how many millions of dollars were lost in the in-out-in-out-shake-it-all-about relationship Motorola had with Symbian.

A Motorola Linux mobe pre-Android

The Linux story is horribly similar: over time, seven different Linux-based OSes were heralded "the next Motorola standard".

Again in 1999, in parallel with all the Symbian and P2K activity, a small team with a semiconductor background could see that Linux was the way forward. The Chinese market wanted handwriting recognition, and an open operating system that wasn't controlled by Nokia and friends, Palm, RIM or Microsoft.

YK Ho, Charles Wang and Ephrem Chemaly started work an a OS called LJ, for LinuxJava, for Motorola. This was much more Linux than Java. Chemaly secured funding for the project from Intel, which was keen to get its ARM-compatible XScale architecture into phones and thus needed a decent OS to run. This was a bit of a coup for Intel, which was up against the rival Dragonball processor from Motorola and Freescale.

The rest of Motorola saw LJ as a niche Chinese product, and Chemaly could see it needed more profile and resources. He went to Ralph Pini, his boss and then CTO preaching Linux as the future of mobile. Pini listened and in 2002 appointed Mala Chandra, who was senior director of Enterprise Java at Sun, to build a Linux team.

The result – trumpeted loudly internally with mugs and mouse mats for everyone - was the Java User Interface Experience, or JUIX, pronounced Juice. This was slow. The JUIX team in Sunnyvale, working with programmers in Chicago, Australia and India, were too used to developing for server environments. Devices with limited processing power, battery and tiny memory were not suited to the software systems they built.

This led to battles between the LJ and JUIX teams, the latter writing cheques its code couldn't cash. Functions generally ran at a tenth of the speed that P2K and LJ, and even they weren't great.

The juice ran out of the Linux JUIX

At this stage LJ-powered phones were selling well in China at high prices and margins; the Motorola Accompli 008 turned up in Europe, and JUIX remained something for the future. LJ could not be killed because it was in shipping products and in the end the JUIX project was written off.

In a political face-saving move LJ was renamed as MotoJUIX and then quietly killed.

A new Linux project was started in 2006 in Denmark and led by the team Motorola bought from BenQ, the Siemens SX1 Symbian developers. Their mission was to build a contractually obligated phone. Kodak had waved its camera patents at all the phone manufacturers and threatened to ban cameras from mobiles unless the technology was licensed. Some paid up, but the Motorola approach was to build a new team, OS and product from scratch and share a cut of the sales with Kodak. It never shipped.

There is some interesting circularity in this as the Danish facility was the place where Dancall built the only dual-band phone to trump Motorola. The design had been sold by Dancall to Amstrad, by Amstrad to Bosch, Bosch to Siemens, Siemens to BenQ, and BenQ to Motorola, which shut it down.

Many of the Java developments were running in parallel; new ones started when an old one floundered, but with committed orders it was often hard to shut down a failing project. And whenever something ran badly more developers were thrown at the software. The learning curve for people new to each codebase slowed everything down.

Meanwhile the hardware people did wonderful things. The strength of the Motorola RAZR was its industrial design, but its huge success perpetuated the myth that Platform 2000 and Synergy were not that bad.

With three attempts at Symbian, vast workforces at P2K and multiple Java teams, things were bad enough, but in each circumstance the teams were internationally distributed. If, for instance, a programmer in Bangalore wanted to know how to use the left or right soft key he'd have to send an email to his supervisor who would then email the person who liaised with user interface team in Chicago who would then email the person in Chicago who understood who wrote which bit of the spec who could then email the UI designer.

The response, which might take a few days while the designer checked with other people, would then go back through the same system so that everyone could be held accountable. If the designer and coder had been in the same place it would have taken minutes - with the Motorola system it took weeks.

The distribution meant that the video people were in Japan, while multimedia was in Italy. There were 31 official development centres with more informal ones not counting the interface designers.

The mismanagement of software probably cost Motorola more than Iridium, it left the company's products lagging.

The sale to Google has the potential to fix the only thing that was really wrong with the product side of the company - the software side. The scale of the redundancies however makes it look as though Google has little regard for Motorola's hardware skills. Google has turned Motorola into HTC. ®

The Register (http://www.theregister.co.uk/2012/11/29/rockman_on_motorola/) · by Simon Rockman · November 29, 2012