The TSB disaster: Where were the grown-ups?

I've spent a couple of years working on IT systems for a large retail bank in the UK, so it is with great interest, and some incomprehension, that I've followed the unravelling of the UK's TSB Bank, now owned by the Spanish Sabadell Group. This organisation has achieved international notoriety, thanks a banking disaster that has cost it almost £200 million, according to its half-year financial results for the six months to 30 June 2018. It caused 12,500 customers to abandon ship, and 2,200 customers to be stung by fraud.

It all happened when it attempted to transfer its banking services from Lloyds to the Proteo4UK platform, used by Sabadell's IT services in Spain. The problems started on the 19-20th May, as the bank shut down services to migrate all customer data to the Proteo4UK platform, and have lasted for months, with issues still being reported, sporadically. Of course, they aren't the first retail bank to hit difficulties during a migration, but normally these last just hours, before a hurried rollback. Here, there was no rollback, despite Lloyds' evident readiness to do so.

There is a lot of public interest in, and a parliamentary investigation into, what happened, and how. IBM, who were called in to help, concluded to UK parliament that "*a combination of new applications, advanced use of microservices, and use of active-active data centers, resulted in compounded risk in production*".

A few of the statements in IBM's brief report are scary to read. This toxic mix of bleeding edge technologies and Agile processes required "*extensive engineering, testing and proving, as well as significant mitigation strategies, including roll-back*" and *'the complexity results in a broad range of technical and functional problems that are hard to diagnose*'. The final sentence was chilling: "*IBM has not seen evidence in the application of a rigorous set of go-live criteria to prove production readiness*."

One must read between the lines of this highly guarded statement. Could it be that the bank had forgotten all its own internal IT rules? Had it failed to learn from other production disasters?

Before the launch of any major change to a banking system, there are extensive tests. On the system I worked on, these alone took six months. After that, there are several trial migrations, for a small controlled subset of the user base, usually the UAT team. Each migration is then rolled-back, to test and refine the rollback procedures, and any issues fixed. Finally comes the launch of the production system, but again this done over a period, firstly just to the IT and UAT team members only, then bank staff, followed by targeted customer groups, existing customers and finally a full launch to new customers.

It seems that none of this happened at TSB. Instead, there seems to have been a lot of attempts to 'roll forward'. The result was an unprecedented melt-down of a retail bank.

I'm not averse to unconventional data architectures, such as Microservices. I like new technologies and have often been considered a 'wild man' in IT. However, in banking processes, I'm a conservative who grips the department's computer manual with whited knuckles. The consequences of error are so dire, and the complexity is so great. That phrase about the TSB, *'a broad range of technical and functional problems that are hard to diagnose*', makes me wince. What works for one sector of commerce doesn't necessarily work in another, especially if the IT staff don't have the necessary training, discipline or experience. It's horses for courses.