Evolving legacies

The latest FStech roundtable, held in association with Delphix, saw panellists discuss the challenges that legacy systems present for financial services, the need for innovation, and the evolving demands that are driving the pace of change

Simon Barrows: Let's begin this evening's discussion by asking what challenges and issues you are all dealing with around legacy systems.

Attendees

Chairman:
Simon Barrows, Director, Archioptiryx

Panellists:
Jes Breslaw, Director of Marketing and Strategy, Delphix
Muhammad Butt, Audit Manager, GE Capital
Iain Chidgey, VP and General Manager, Delphix
Sandeep Chugh, FX IT Development Manager, Credit Agricole CIB
Paul Comber, Head of IT Risk, HSBC
Halldor Fosså, Infrastructure Architect, RBS
Rob Handicott, Chairman, BCS Financial Services Group
Harish Khatri, Head of Operations and CTO, Axis Bank UK
David Layton, EMEA Financial Services Director, Delphix
James Macleod, Project Manager, Compliance IT, Standard Bank
Peter McElwaine-Johnn, CTO, Aldermore Bank
Ian McLean, CIO, Sterling Technologies
Stephen Murgatroyd, Business System Designer, BCS Financial Services Group
Glenn Murphy, CIO, London and Capital
Andrew Veitch, Director, Anthemis

Halldor Fosså: At RBS we are in the process of moving from several in-house legacy CRM systems to Microsoft Dynamics. The incentives for doing so are obvious; reaching out to customers and trapping a single view of that customer is key to any business. But the challenges around this type of migration include data platforms, bringing users' desktops up to the required standard, and the governance of data. And we are doing all of that in the context of a rigid budgetary climate and ongoing business change.

Muhammad Butt: I think you can break down legacy system issues into two buckets. One is identifiable legacy – when you know you have got something out of date and need to do something about it. The other is when things pop up unexpectedly – for example, if you weren't aware that a change you put in production 10 years ago would effect a change you are trying to make now. That type of thing can often cause the biggest issue.

Glenn Murphy: In terms of innovation, I question whether the problem is the system or the data. I think that data is the real issue. Whatever system you have, it's about getting data from it that's in a good state, is clean, and has a strong degree of integrity – the innovation is the system that just interfaces to take that data out or stage it somewhere in a new form. So is it the legacy system or the legacy data – which is key for innovation?

Ian McLean: I think it's a combination of the two really. One of the main challenges we have with potential legacy is the ability to change and then successfully test systems at the rate the business needs. Another is the technology side, and whether the technology is becoming obsolete and no longer supportable or maintainable. If a supplier pulls out of supporting a technology then you obviously have a decision to make.

Sandeep Chugh: I think it varies from system to system. Sometimes people [in the business] are really tied to a system and are reluctant to upgrade. And while we can change the database

behind the scenes – and there is often no issue in that – it's difficult to change the upfront system because people are still very wedded to it.

MB: It's also important to consider that while we have identified obsolescence as being a key driver of system breakdown and control problems, almost all of the major control breakdowns that have made newspaper headlines were due to failing BAU processing and a lack of adequate governance and oversight – when someone had access to systems they should not have had access to in the first place, and no one bothered to check or challenge that.

SB: There are often a lot of different and conflicting demands driving a systems change, whether that be compliance, launching a new product, or the need to innovate. What are some of the competing change demands that people have experienced?

James Macleod: It can be difficult when a system has a lot of technical debt that builds up over time and eventually becomes a legacy system. The challenge is to get business justification to deal with the problem while it's manageable, and prevent it becoming a legacy system. However, there are often other pressing business demands which are prioritised because they affect the bottom line, and before you know it, you have a mammoth legacy system which no one wants to touch and is very difficult to change.

Rob Handicott: It may be that there is no single person within an organisation who has been through all of the changes in a legacy system and can handle it in-house easily – that is certainly one of the problems the industry faces. And there might not be the documentation to say how and why those changes were made.

Stephen Murgatroyd: If you look at new entrants in the banking sector, they are going for brand new core banking systems. But that does beg the question of whether in 15 years' time from now those systems will still be flexible, as they are not necessarily that well documented because they have been brought into existence very quickly.

Iain Chidgey: If a new bank started up their systems today they would have no customers on day one, but in 15 years' time they would have many customers and a whole lot of data, and that is the data they will want to harvest and mine to keep them in business and do more things with. They might make system changes in an incremental way because they have got the opportunity to do that and are agile, or they might just put a system in place and wait until it gets to creaking point, and have to go for a 'big bang' change approach. Looking at the case of Deutsche Bank at the moment, they are going through a big transformation and innovation project. There are billions of

Room with a view: The roundtable took place at the Gherkin

dollars of cost that they want to strip out, not just from IT but in the way they are reforming the business as well. A lot of companies are going through this type of change, and it is driven by the business, not technologists wanting to move away from certain systems.

IM: I'd also say that data is nothing without the business logic telling it what to do. In a lot of these older legacy systems you have up to 25 years' worth of business logic around how things need to be done for the regulator, or for the downstream and interlocking systems. So moving the data out is great, but you have to consider the 25 years' worth of intelligence that has built that system.

SB: There are so many factors driving the need for change, whether that be mobile innovation, customer demands, compliance, or the need to take cost out of the bottom line. Is it the case that this demand for change is constantly increasing now?

> "In terms of innovation, I question whether the problem is the system or the data"

HF: The business is always going to want you to be able to innovate faster – that means everything is inherently legacy because you can never satisfy the need for change fast enough.

Peter McElwaine-Johnn: I think legacy is what we call a system that can no longer sustain the rate of change that we need it to. And with current accelerating rates of change, that means things are going to become legacy younger and younger.

MB: There is also the security threat to consider. As products go towards obsoletion and unsupported versions then you have to upgrade them, otherwise you are exposed to a high security vulnerability. If you get attacked once, and that becomes known, everyone is going to try to exploit that vulnerability overnight. Of course, when there are a large number of systems running together with a large number of users in the business, you can't just pull the plug. So it may be that a business is stuck with that vulnerability for the next two or three months while they try to turn it around.

RH: Security and sustainability are certainly very important issues going forward. Hopefully lessons can be learnt from old systems, and newer generation systems will be able to accommodate these kind of issues much better.

Jes Breslaw: Isn't also part of the problem that there is a lot of fire-fighting going on, as well as siloed projects? One part of a financial institution might be working on a compliance project, while another one is working on modernisation for mobile, for example. But there is actually synergy between those, and if you used a stress testing project to make data available more quickly, there would be a direct benefit. A more horizontal modernisation could have a big impact on financial services that are currently sitting within silos in departments.

David Layton: I think there is another challenge that hasn't been touched on yet. The vast majority of financial systems in large organisations are still based on a Z-series mainframe; but while people with Z-series skills are leaving organisations, university students aren't interested in Z-systems. So it's going to get to a critical point or crossroads where businesses are not going to have people with the skills to access data or change legacy systems.

SB: On that point, what are some of the other hurdles and constraints when delivering change?

IM: Quite often you have a choice to make between the integration and migration of systems. Sometimes it's not possible to integrate because it's a closed system, you don't have the code base, or the package provider is not supporting it anymore.

HF: I previously worked on the Aspire contract, which was then the UK's largest IT outsourcing deal. The one key word that we used over and over again was legacy 'erosion', because it was recognised that some of the huge mainframe legacy could not be 100 per cent turned off. So in terms of migrating it, we looked at identifying individual, valuable bits of data, and gradually extracting those columns or tables to move into a new world.

SM: You make an excellent point, and banks in particular have been trying to address this for a number of years. A lot of them have been going down the 'data bus' route, where data that cannot be enhanced in a legacy system is taken out and put in a data bus, along with new business data. Eventually there is more stored in the data bus than in the legacy database, and the old system is not used as much so it dies a natural death, without a big switch off or switch over. So it is what you might call a graceful degradation – or erosion.

PM: I think it's often the complexity of IT that opposes us every day, takes away our agility and stops us from being able to change things at the pace that we want to. I've done quite a lot of research on this, and there are really three types of complexity. There is necessary complexity – the minimum amount that you need to have in the solution domain. Then there is unnecessary complexity – what is added into the solution domain through bad design decisions, but that doesn't need to be there. And the third one is age-related complexity. No matter how simple something is when you build it, if you wait long enough it becomes complex, it wears out, you lose systems knowledge, or it no longer complies with security standards, and you then have massive problems changing it. If we focus on the unnecessary complexity and the unnecessary dependencies that we introduce between things, that is where we have the opportunity to make the most difference.

> The accelerating rate of change means that things are going to become legacy younger and younger"

IC: Whether you move to a Data as a Service platform, or whether you do it the old way of manually provisioning hundreds of copies, I think the point is that you can't innovate unless you have an open mind to overcoming these unnecessary complexities. You have to be constantly looking at new ways of doing things and be open to making changes, otherwise we don't evolve.

SB: We've covered several angles today, and I think that living with legacy system complexity as best you can, and dealing with it better than everybody else, has to be something that everyone strives for. We should all be embracing any tools, techniques, abilities and plans that enable us to live with this complexity, whilst trying not to introduce any more of it in the changes and innovations that we make.

Delphix DaaS (Data as a Service) software helps application projects move faster, with higher quality, and at a lower cost. More information at: www.delphix.com