No Code/Low Code (NCLC) Technologies – what's all the buzz about?

Altus

MAKING FINANCIAL
SERVICES FASTER

In association with

fintechOS

Altus Consulting is a specialist provider of consultancy services to the Financial Services sector.

We help clients achieve operational excellence and improved returns through a combination of market insight, proven industry models, and technology expertise.

For more details of these services please visit our website altus.co.uk

Contents

Introduction

Every business wants to grow its revenue and reduce its costs. In the highly competitive Financial Services market that inevitably means smarter use of technology; whether that's to engage more customers or to improve the efficiency of core administration.

However, our industry is not noted for its early adoption of new technology. Gartner noted (Gartner, 2018) that the Insurance industry lags behind other sectors and is either at an early stage in digital maturity or tends to follow a cautious approach to digital transformation. This is perhaps understandable for a highly regulated industry, laden with legacy systems and a long standing reliance on manual processes. With the increasing demand from clients for more timely and frictionless models, this pattern needs to change. The good news is that the new breed of 'no code / low code' (NCLC) tools focusing on Financial Services now delivers on that promise and is ushering in a step change on how providers can deliver new digital enterprise applications rapidly.

How can Financial Services providers accelerate digital transformation, to not only catch up with other sectors but also to drive ahead at rapid pace? Different industry analysts will point to their own favourite categories of technology in an attempt to answer this question but one thing which is clear is that digital transformation will rely on the ability to deliver applications. These range from direct customer engagement channels such as web portals and mobile applications, through automation of internal workflow, to industry-specific tools for claim management. There is a clear business advantage to delivering these applications into production environments rapidly, so that they can deliver commercial benefits quickly, rather than the first-generation web technology that is rapidly becoming the new legacy.

This new landscape interacts with external services to simplify the customer experience and ensure greater efficiency, accuracy and decision-critical information for the Financial Services provider.

Low Code (LC) technologies promise a shortcut compared to traditional development tools and coordination of components from a 'colony' of specialist providers servicing the Financial Services sector. The underlying premise is to reduce the need to write lines of code by re-using prebuilt components. The applications are commonly constructed by dragging and dropping these components via a Graphical User Interface (GUI). In this way code is abstracted away from the application developer to the point that they may even remove the need to write code at all. These environments are called No Code (NC) platforms, but for the purpose of this paper, will be included in the overall LC category. You can think of them as dragging components into the cells of a preconstructed honeycomb (the NCLC platform) to rapidly create a functioning system.

However, a note of caution; we have been here before. Over the last 30 years first 4GL, then CASE and then RAD tools were all touted as the nirvana of application development. None of them ever quite delivered on that promise so, what's different this time round? Well, a global pandemic that has forced the overnight adoption of digital services is certainly one factor but there are others too and this white paper sets out to explore them.

Find out more about:

- The evolution of LC technology and the latest state of the market

- How new layers of specialised Financial Services components speed up development

- The challenges of LC adoption that still need to be overcome

- How some Financial Services providers are using LC technology in practice.

By the end, we hope you'll be in a much better place to decide whether LC technology could work for your business and which suppliers you might want to talk to about their toolsets. Of course, we'd also be delighted to help you on the journey…

Background and Definition

What is code, and what is Low Code?

Computers are fast, but dumb – they need to be given instructions telling them exactly what to do in any given situation. Those instructions are provided via a special coding language – typically using some words and concepts from human languages, but with their own very specialised rules and structures.

Traditional software development requires a high degree of specialist knowledge and technical skill across a wide spectrum of roles e.g., low-level software engineering through to application development, operational deployments, and testing. Can this complexity be reduced in any way? This question has been the driver of the LC paradigm. If the amount of code can be reduced to as little as possible, perhaps even to zero, it should be possible to deliver applications quicker, more reliably and with a significant cost saving.

Evolution of the NCLC Paradigm

LC application development is not a recent innovation. The goal of making the development and coding of business-specific applications accessible to non-specialist programmers has been addressed over time as programming languages evolved. Functionality would be based on pre-existing libraries hence reducing the number of lines of code needed to perform powerful data analysis and visualisation.

Subsequent technological developments resulted in the more widespread use of the Graphical User Interface (GUI), and the origins of what we have come to associate with modern LC platforms i.e., constructing a graphical picture of an application or workflow using Drag and Drop tools and the automated generation of packets of code, versioned controlled and backwards compatible with new releases.

Business Process Management (BPM) vendors were quick to utilise GUI advancements to enable the definition of workflows in graphical modelling tools. The visual definition of a workflow would translate to code representing a business-specific model, ready for deployment to the enterprise workflow engine. Business users now had the ability to define and generate automated workflows, and automation of workflow remains one of the key applications of Low Code platforms.
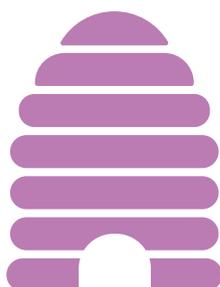
But the use of the graphical editor in a LC platform is not restricted to business process definition. Traditional data transformation and integration platforms, for example the likes of Talend, Informatica or Mulesoft, have evolved with the graphical editor being central to generating automated processes for business-critical data transformations and service orchestrations. The level of technical as opposed to business knowledge, being much more significant in these examples.

So, considering the relatively long and admittedly chequered history of attempts to simplify application development by de-specialising and standardising tools and methods, why have they become the popular and exciting technologies now and in the future? Consistent through the evolution of LC frameworks is the central tenet of the accelerated development of applications, via iterative cycles and rapid prototyping. This aligns with the Agile development process which is now a widely adopted methodology within organisations today. Also, business applications can be built by non-specialist "Citizen Developers". It is the ability of these Citizen Developers to democratise system developments that has revolutionised the step change in productivity and reduction in cost. Now any 'worker or drone' can manifest beneficial ideas that can instantly be used by the rest of the colony of colleagues where beforehand such ideas would often get lost in endless roadmap and governance committees. So, have current LC tools learned the lessons of the past and are these aspirations really achievable?

Compared to 40 years ago we are in a different world of accessible technology in terms of computing power, graphics, functional libraries, services, cloud resources, external services and data. Also, the growth of the internet and the demand for Web and Mobile applications has expanded the original boundaries perceived for LC application builders. This led to the rich technologies and diverse ecosystem of LC technologies we see today.

The challenge is to match the right technology and provider to business cases relevant to Life and Pensions (L&P) and General Insurance (GI) organisations today. This is critical to ensure organisations do not waste time trying to research the plethora of providers unnecessarily. Access to non-public information and experience of working with providers through systems implementations that an independent consultancy can provide is essential to ensuring success for Financial Services providers.
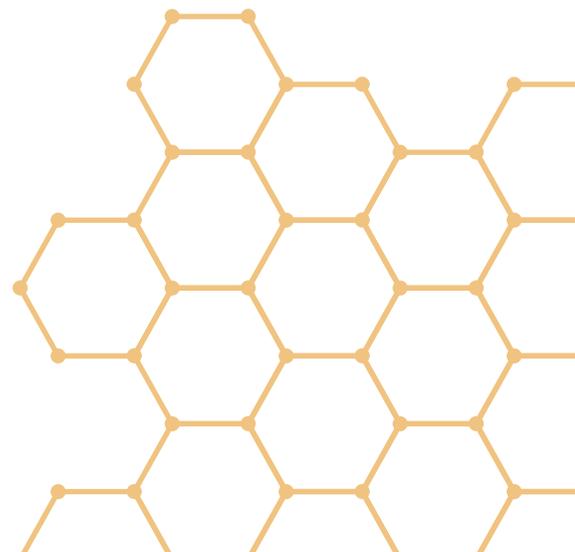
Pros and Cons of Using LC Principles to Build Applications

Potential Benefits

The evolution of LC applications described above, highlighted several appealing benefits to the Financial Services industry. These are summarised as follows:

Increased speed of delivery and reduced development costs:

removing coding effort accelerates the delivery of applications to the business. This reduces the cost of delivery compared to a traditional software development cycle and enables ROI to manifest sooner. LC application delivery lends itself to the Agile process and "Fail Fast" philosophy as it significantly reduces the time involved in completing the repetitive yet critical stages of software delivery (build & deploy), meaning that iterations and prototyping can take place more frequently.

Baked-in Functionality:

fundamental to LC development is the re-use of prebuilt components. In effect, the difficult work associated with coding has already been done. For example, not only is complex code writing either eliminated or reduced, but the effort of maintenance, core testing and security has already been expended. We shall see that there are many pre-built components tailored specifically for the insurance provider domain which, if chosen appropriately, can rapidly progress the digital maturity of an organisation.

Maintenance and currency of browser standards & App stores:

An attraction of LC platforms for providers requiring omni channel digital UI is the ability to deploy apps that should be pre-approved by the Apple & Android stores. Not only can this accelerate omni-channel adoption of the solution but should also mean they remain updated as long as the LC vendor commits to keeping up with any changes the app stores require. This can save significant time, testing and expense.
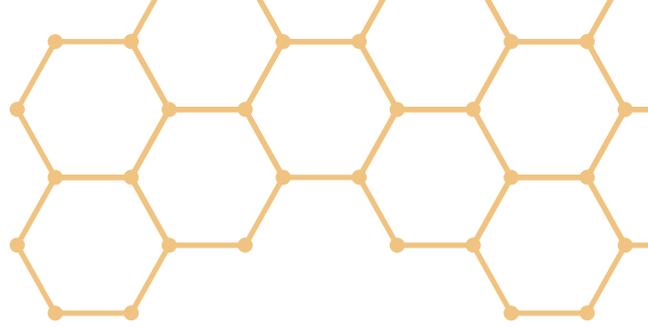
Cloud hosting:

LC platforms are typically cloud hosted system, and therefore provide the associated advantages of not having to determine your technology stack, 'ready to go' or 'try before you buy' services, resilience & recoverability, scalability, security, access to complementary hosted services e.g. AI bots with market specific skills. They are usually classified as Platform as a Service (PaaS) solutions since, not only are the development tools provided, but also a managed execution environment for the resultant applications. Thus, there is provision for end-to-end delivery of applications from concept to production.

Involvement of Citizen Developers:

with less need for coding there is less IT specialised expertise required in software development. This enables greater collaboration between non-technical business users and IT stakeholders. The term Citizen Developer is used to describe such non-IT business specialists involved in using LC platforms to deliver applications which can reduce the need for rework and iteration due to misunderstood requirements. LC systems designed to support these non-IT users have a significant advantage over their competitors as they will be able to deliver, faster benefits across by leveraging a wider user base across the colony of specialists in the Financial Services provider.

Reduction of Shadow IT:

organisations have more control over applications developed using LC than customised End User Compute (EUC) applications, e.g., spreadsheets. Citizen Developers developed their own solutions in the past because they had no other choice. Now, using similarly simple interfaces, they can develop more extensible solutions. The solutions can be reviewed and maintained with more quality control and oversight than was possible with EUC.

The reduction in shadow IT

is further enhanced as it is standard for LC solutions to provide a wide range of connectors to integrate directly with existing systems and data sources. As such it would no longer be necessary to take feeds from data sources and manipulate with business logic in applications falling outside the realm of IT governance.

Caveats and Challenges

There are many appealing qualities of LC development for Financial Services enterprises but there are some challenges to consider. These can be categorised as intrinsic issues of taking a LC approach vs strength/weakness areas to watch out for when choosing a vendor.

Intrinsic Issues

Fragmentation of IT estate:

Adoption of multiple LC solutions may result in a fragmented landscape of platforms across an IT estate. This can result in siloed development patterns and potentially, expensively acquired applications quickly falling out of favour and not being used. The ability of these platforms to integrate with each other is vital is architecting an effective enterprise IT.

Governance:

the delivery of applications via LC technologies should be subject to the same governance as traditional software development. Consequently, there should be adequate governance oversight, quality control and audit measures.

Management of Custom Source Code:

code that is written by developers using a LC platform, whether they be specialist or citizen developers, needs to be managed. Ideally, this code will be manged through a specialist version control tool. However, there is the danger, especially with non-specialist developers with no experience in these best practices, that code will escape this governance and become distributed throughout the building blocks of the application. This makes bug-tracking difficult and can lead to duplication of code in many places.

Shadow IT:

whilst LC platforms help Citizen Developers to rapidly produce new systems, there is also a danger that without the right governance they can produce applications that aren't compliant with internal policies, tested or documented.

Vendor Selection

Performance and Security:

Although "baked-in functionality" is a plus point for LC platforms, ultimately, you are running someone else's code. In this respect control over issues like security and performance are in the realm of the provider. Questions such as "Is the application secure and compliant with InfoSec policy" or "How can we improve the speed of this application" may therefore be difficult to resolve.

Vendor Lock-In:

Once investment in a LC platform has been made it is important to maintain its value to the organisation. There will potentially be expensive annual support and maintenance contracts and by tightly coupling custom source code to a specific provider's framework it may be necessary to replicate coding effort if different LC platforms are used in future. Also, if the vendor falls behind the curve of staying up to date with competitors, then the value to the user will diminish over time.

Intended Audience:

some LC systems are designed to be used by back-office admin where the UX controls are appropriate. Some platforms can work headlessly, so corporate UX guidelines can be applied as the presentation layer. If the apps are intended for use by consumers, then a richer and more engaging UX might be required. Caution should be taken to fully understand the fit to your audience's requirements when selecting a LC platform.

Pros and cons of Low Code / No Code

Pros of Low Code/No Code

- Speed of delivery
- Baked-in Functionality
- Standards
- SaaS
- Citizen Developers
- Reduction of Shadow IT

Cons of Low Code/No Code

- Vendor Lock-in
- Fragmentation of IT
- Performance & Security
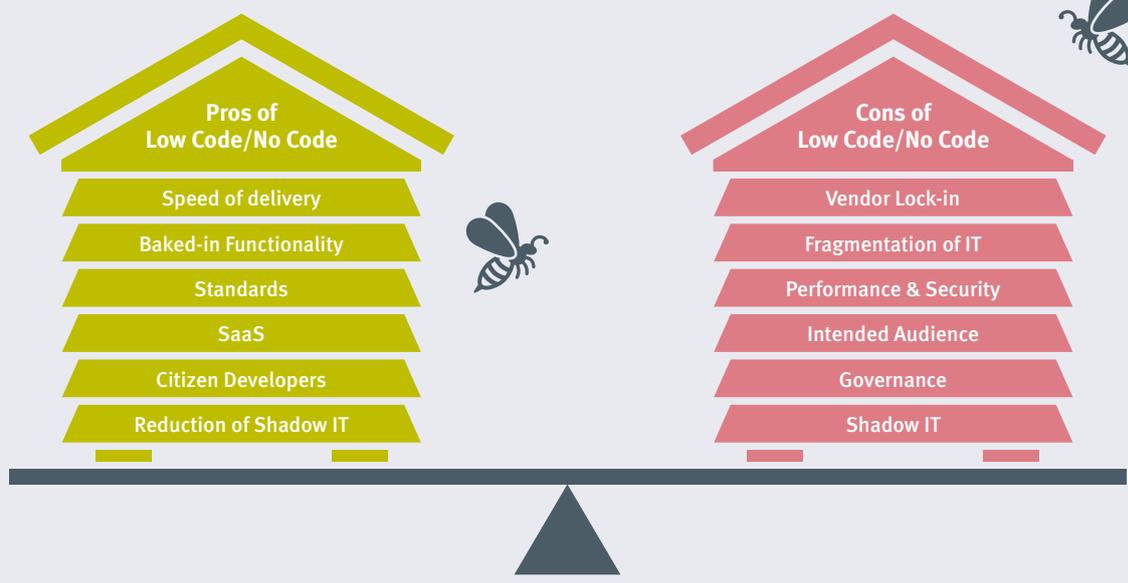- Intended Audience
- Governance
- Shadow IT

Figure 1: Summary of LCNC pros and cons

What is LC today?

The evolution of LC has resulted in a diverse pool of providers and platform types. They range from purely NC solutions through to platforms of higher complexity which expose the ability to add custom code. This spectrum is key to understanding which providers and solutions are appropriate for solving business problems, since the line between what is termed NC and LC can become blurred. Understanding which providers also have experience and specific capabilities in your specific market in addition to generic functionality, is also important to identifying the right shortlist.

NC Platforms

There are many providers marketing themselves as NC and the choice can be bewildering. The nocodejournal.com organisation has matched a total of 268 separate NC providers to eight categories of applications ranging from Web and Mobile applications through to the generation of automated workflow and chatbots (NoCodeJournal, n.d.) organisation (Table 1). This collection is by no means complete.

Navigating through these options can seem daunting but the sheer number implies a real market demand. Even the large cloud service providers have noticed this and provide their own solutions, e.g., in 2020 AWS launched Honeycomb and Google acquired Appsheet.

Total numbers of individual NC providers

Domain	No. of Providers
Web Application	76
Business Application	34
Mobile Application	39
Automation and Workflow	74
Payment	13
Chatbot	10
Database	15
Other	6
Total	**268**

Table 1: Total numbers of individual NoCode providers based on Business Domain cited by NoCodeJournal https://www.nocodejournal.com/state-of-nocode

LC Application Platforms

LC platforms retain the characteristics of their NC counterparts, i.e., a graphical interface for designing an application with pre-built components, but with the ability to program extra functionality. Hence, there is a need to have some familiarity with coding. In fact, it is possible that experienced software developers will already be using LC platforms in their daily routines, exploiting the ability to produce applications directly.

There are LC platforms which are generic to the point that they can be used to develop any type of business application, ranging from portals and mobile apps to intelligent automated workflow, and include leading names such as Appian, Mendix, Salesforce and OutSystems. This group of solutions are typically termed Low Code Application Platforms (LCAPs).

In general, applications created by LCAPs will represent a step-up in criticality to the business. For example, it may be necessary to define and execute custom business logic and/or amalgamate complex data through different interfaces ranging from new Application Program Interfaces (APIs) to legacy systems requiring specific connector interfaces. However, they are all pitched at increasing development productivity through use by citizen developers as well as technically skilled software developers.

NCLC Technologies For Insurance

The previous examples have described technologies that enable users to build powerful applications from the ground up using generic, but rich, pre-built components. Other technology providers have taken the principles and methodology of low code to provide platforms or solutions specific to the needs of the insurance sector. They have invested the effort into constructing tools and solutions relevant for use in standard Insurance business processes, such as claims management or quote and buy journeys. End users can enhance out of the box functionality using in-built Low Code, sometimes completely No Code, tooling features.
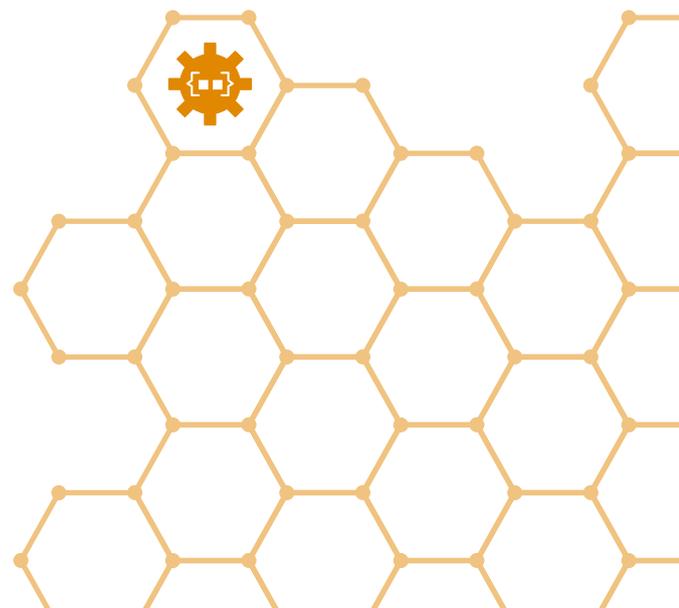
How is it being used in Financial Services?

Z ZURICH

Zurich Insurance chose the Mendix LCAP to empower the UK DevOps team to deliver applications using Agile working and iterative development (Mendix, n.d.). The DevOps team was comprised of non-specialist developers and some Business Analysts, since putting together a software development capability from scratch was deemed too impractical. An initial project was to increase the efficiency of an existing legacy application helping provide customers protection against loss incurred by acts of terrorism. The existing application interfaced with re-insurer Pool Re and needed to comply with its underwriting manual for details on premiums to charge, discounts & adjustments, when they must send data and in which format. They replaced the legacy system and built the Terrorism Data Capture (TDC) application, improving data integrity and the user experience for Zurich's 700 underwriters. Key benefits of the LCAP were the out of the box capability for continuous integration & delivery and the ease of integration to existing enterprise systems (such as the Zurich Salesforce instance) using prebuilt connectors.

XCEPTOR

Xceptor, for example, is an enterprise NC automation platform that enables its financial services organisations to simplify data ingestion, transformation and digitisation of processes. The automation process is defined by business users through connecting prebuilt components to take data from numerous sources through to a state ready to be consumed by downstream channels and other processes. An analysis by Forrester calculated an overall ROI of 267% and a payback period of 6 months from the use of Xceptor from four Financial Services organisations.

Miller, an insurance firm used Xceptor to significantly reduce the amount of manual effort in collating data from multiple complex spreadsheet formats and make the process significantly less error-prone (Xceptor, n.d.).

A common theme for business specific LC vendors is to reduce the dependence on specific technical skills for the financial service providers' staff. As we have seen, data originates from multiple sources, many of which will be legacy, and all have some contribution towards making astute business decisions. Whereas LC solutions are available to automate back-office processes, there is also a need to present disparate data visually to the end user, commonly using a dashboard.

dt dunstan thomas

Dunstan Thomas's LC solution, Integro CX, has been tuned specifically for integration of financial data sources from multiple platforms, providers and generation of sector specific KPIs. The platform was used to build portals for Discovery Group, a major insurance company established in South Africa and now with a global presence and 5.1 million customers. Tailored portals were built for back-office users and advisers for their business and client management. Key to this delivery was the ability to integrate over 60 APIs in a reusable way, highlighting the value in speed of delivery and prebuilt integration (DunstanThomas, n.d.).

Some InsureTech providers have taken the opportunity to build complete digital insurance solutions around a LC framework. They utilise domain specific business knowledge of the sector when building the core applications and provide the ability to customise its' design, build, launch and maintenance through LC.

fintechOS

FintechOS – as the name implies FintechOS are a LC vendor that have chosen to focus on the Financial Services market developing their platform to deliver rapid benefits to providers in the General Insurance (GI) and Life & Pensions markets. They specialise in knitting together the tapestry of data sources, legacy systems and components needed to deliver rapid digital solutions.

For example, OMNIASIG Vienna Insurance Group, sought a partner to develop their OMNI+ Health Insurance Platform for their new line of business - group health insurance, a solution that tackles underwriting, distribution, an extensive network of healthcare providers and end-to-end claims management. Omniasig needed easier, faster and more transparent processes, digitised customer facing processes and so designed innovative services that would deliver added value to their customers. FintechOS acts as an interface between their legacy core systems, third-party solutions and apps, and as a modern digital layer, Omniasig uses FintechOS to accelerate digital transformation and create frictionless, personalised customer interactions and experiences. With FintechOS technology Omniasig is able to do this development in-house and at a fraction of the cost of traditional development services (FinTechOS, n.d.).

The benefits realised by the new system include a 75% decreased sales cycle time, a 50% increase in operational efficiency, 65% improved customer experience all delivered through a 100% digital onboarding omni-channel solution.

INSTANDA

INSTANDA offer a fully customisable cloud-based Software as a Service (SaaS) policy administration platform aimed at both carriers and Managing General Agents (MGAs). Working with the client in initial stages they claim typical implementation durations to be between 8-12 weeks. An appealing metric when compared to building from the ground up using traditional software development cycles. Insurers can then create and test new products quickly, make changes according to feedback, and have assurance of compliance with regulatory needs. INSTANDA, in partnership with a consultancy, delivered a solution for a top 5 global insurer that significantly improved the efficiency of one of their most complex speciality books (Instanda, n.d.). In just eleven weeks, they applied the INSTANDA platform to digitise complex processes which will enable the insurer to increase the size of their book by 100% in three years.

CANDELA LABS — AWPL Reborn

Candela Labs are keen to stress the importance of providing pre-built components and their ability to integrate with existing systems by moving beyond the label of LC to a definition of an Intelligent Automation "Fabric". The aim being to weave around the fabric of existing systems as easily as possible. Their Intelligent Engagement application suite has been created specifically for Life, Health and General Insurance, with pre-built internal and customer facing journeys across virtual assistants, portals, mobile applications, and unified messengers. Automated processes are designed using the accompanying Intelligent Operations suite. Coding is an important part of enhancing out of the box features and templates, but the source code management remains well separated from the suites, avoiding concerns of un-managed code spread across applications. Candela Labs was used by a Malaysian life insurer to fulfil a requirement to demonstrate to a regulator an automated business system within 60 days, as part of a new entity acquisition process. Using existing templates for core processes, additional processes needed to be developed for data entry and underwriting. The new business solution was fully integrated with existing systems and deployed within 7 weeks. While the system was deployed under a regulatory deadline, it is still being used "As-Is" after two years with minimal modifications.

The NCLC Honeycomb of components

A colony of bees is a perfectly evolved single entity that only exists as the sum of its parts. An individual drone wouldn't survive on its own, nor would the Queen. The honeycomb, like the LC platform, provides the structure for the Workers to develop and grow the community, housing all they need to survive and thrive.

In the same way, the successful Financial Services provider's digital solution, is the sum of its parts orchestrated by the LC platform. It will consist of ID&V components, AI fraud detection, AI unstructured data to digital ingestion tools, pre-configured workflows, etc.(Figure 2)
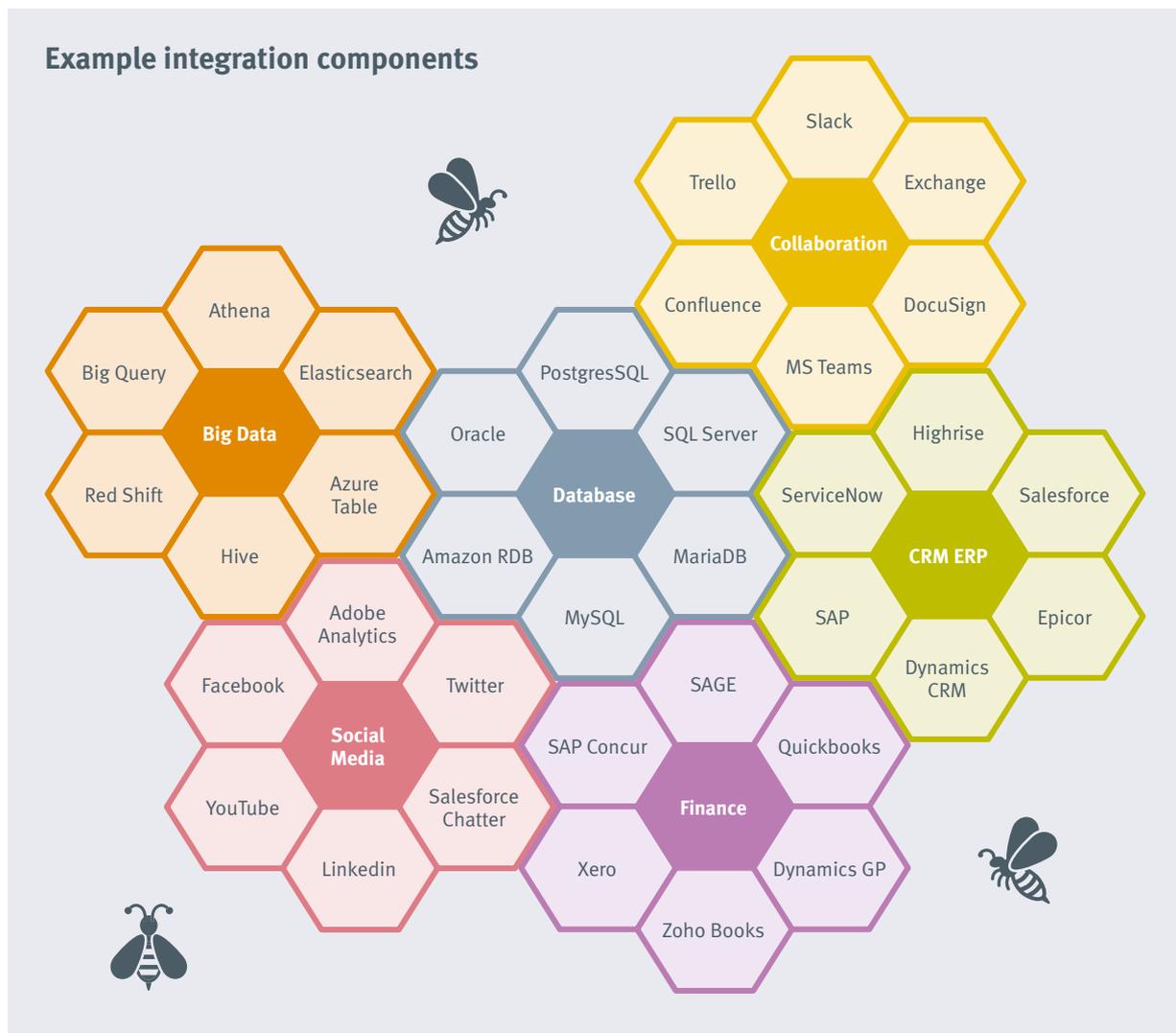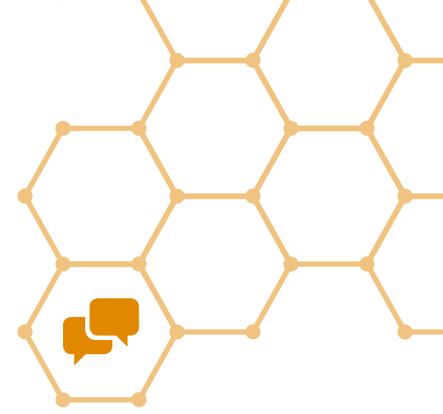

Figure 2: A subset of integration components for the FintechOS platform. The honeycomb pattern highlights the robust management of integration points by the platform itself.

Discussion Points

Where is the value?

For any organisation, the benefits of adopting LC technologies must result in a convincing cost-benefit ratio to warrant support and investment. The pros and cons of investment can become complex once business requirements are assessed against factors such as budget and existing digital capability. How can we evaluate LC technology adoption to determine if the investment is attractive and which platform provider will deliver the greatest benefit?

So far, we have highlighted the ability of LC platforms to deliver applications more rapidly than the conventional software development lifecycle (SDLC). Figure 3 shows a hypothetical comparison of change projects using both methods for delivering similarly scoped applications. For the SDLC, costs quickly rise to a fairly constant level until the point of delivery into production. In contrast, the first project using a low code platform ramps up to a higher cost level, covering initial investment in the technology (consultancy, training etc), but achieves delivery more rapidly than the SDLC. Subsequent low code projects can start sooner, starting and continuing potentially at lower costs since the number of developers needed to build an application from pre-built components would be less that the conventional SDLC.
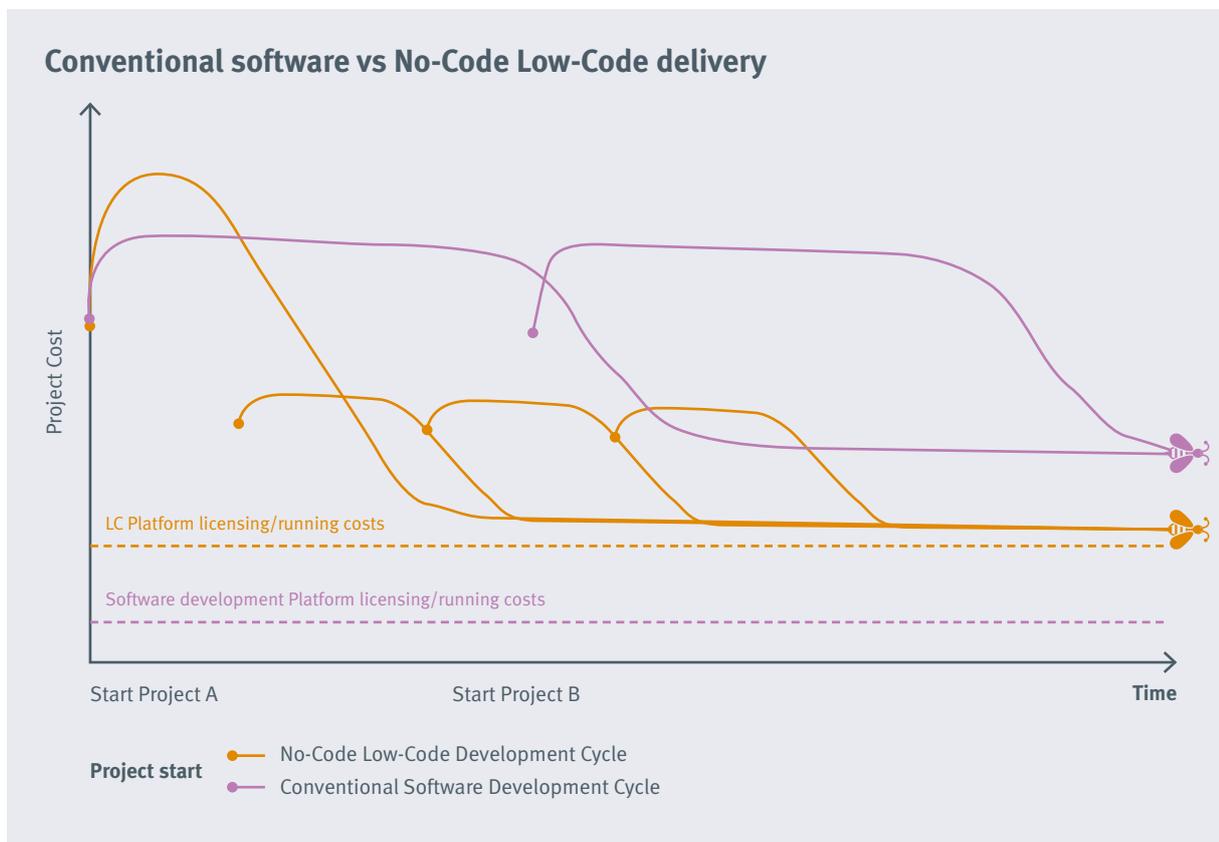

Conventional software vs No-Code Low-Code delivery

Project Cost

LC Platform licensing/running costs

Software development Platform licensing/running costs

Start Project A   Start Project B   Time

Project start
— No-Code Low-Code Development Cycle
— Conventional Software Development Cycle

Figure 3: A comparison of cost trajectories for two projects using either conventional software development delivery or No-Code Low-Code delivery.

Discussion Points (Cont.)

However, the curve trajectories of the projects alone do not provide a representation of or total cost, since the overhead of technology licensing and running costs, including bug fixes, enhancements, upgrades etc, needs to be incorporated. Figure 3 highlights that the running costs for Low Code platforms is likely to be higher than, for example, maintaining the tools of a software development department using open-source technologies. The combined cost of change projects and running costs can influence the viability of LC platforms for an organisation and there are a number of factors to consider. Figure 3 clearly shows that LC platforms suit organisations where change projects are frequent and regular, but importantly, concentrated in the applications provided by the LC platform.

Having decided that a LC Platform would bring cost savings to an organisation, the next step would be to ensure the correct choice of vendor. We have highlighted that low code platforms can deliver applications rapidly and are therefore suited to organisations with a high frequency of change and backlogs which grow quickly. A compounding factor in determining which platform is most appropriate is the complexity of change or transformation. Complexity usually increases with the specialisation of use cases and with the number and intricacy of integrations. Implementation and subsequent changes to a new insurance Policy Administration System for example, would suit a low code platform from an insurance industry specific technology vendor, who have already built components for insurance use cases, and which specialise in integrating with financial systems.

There are many factors that need to be considered in determining whether a LC platform fits a particular organisation's requirements. After the initial excitement of investment in an expensive LC platform and delivery of the first application, will there be a constant pipeline of business cases that can justify continued subscription to the service? Can your organisation continue to spend money on training business and technical employees on

the LC specific skillsets? Are specialist low-code developers more or less expensive than .Net or Java developers? What is included in the licensing model? Overall, the LC platform needs to uplift an organisations agility in providing a business response to volatility and disruptions in the insurance market to justify the investment.

Layering of Financial Services specific functionality

So far, this paper has emphasised the wide range of LC technologies available to rapidly accelerate the delivery of agile digital solutions for Financial Services providers. The options span from generic solutions which enable applications to be built from the ground up using pre-built components, to enterprise platforms designed exclusively for Financial Services core business cases. Clearly the choice of technology will be determined by its resulting value to an organisation and the effectiveness of the "Horses for Courses" choice is dependent on a clear view of the LC technology landscape. However, what has become clear is that those NCLC vendors that have chosen to focus on a vertical market like Financial Services are starting to carve out a niche for themselves by delivering more rapid deployments tailored specifically to their market. This in turn delivers greater value to Financial Services providers as the cost of bespoke integration into old legacy systems or market specific data standards, is taken care of by the NCLC vendor.

For example, NCLC vendors who understand the landscape they are targeting have written specific adapters for legacy admin systems that aren't API enabled. This allows them to offer similar benefits to other 'stranded' Financial Services providers who find themselves on the same admin platform. This means the traditional approach to decommissioning an old legacy admin system (i.e. migrate policies in run off to a new digitally enabled one), would fail a business case for books of business with less than 10 years to run off, because the savings wouldn't recoup the costs over that short period of time. There is now an

alternative option for this scenario. Using NCLC platforms to deliver digital self-service capabilities in this use case is considerably more likely to meet the requirements of a business case - especially if the Financial Services provider has a number of stranded legacy platforms it wishes to digitally enable, and which can share the NCLC platform's license fees.

It is this market specific focus and layering of benefits that NCLC platform vendors such as FintechOS have identified as a rich hunting ground, rightly so, as so many large life and pensions providers have 'legacy' or 'heritage' books crying out to be brought into the digital age. The often 'swivel chair' processes are manual, hugely expensive, unscalable and represent significant risk of errors. Transforming them with NCLC platforms can breathe new life into a provider.

This layering of value is analogous to a beehive where the value of the honey increases as more 'supers' are added by the beekeeper. The greater the UK Financial Services focus the greater the value to both the NCLC vendor and the Financial Services provider.
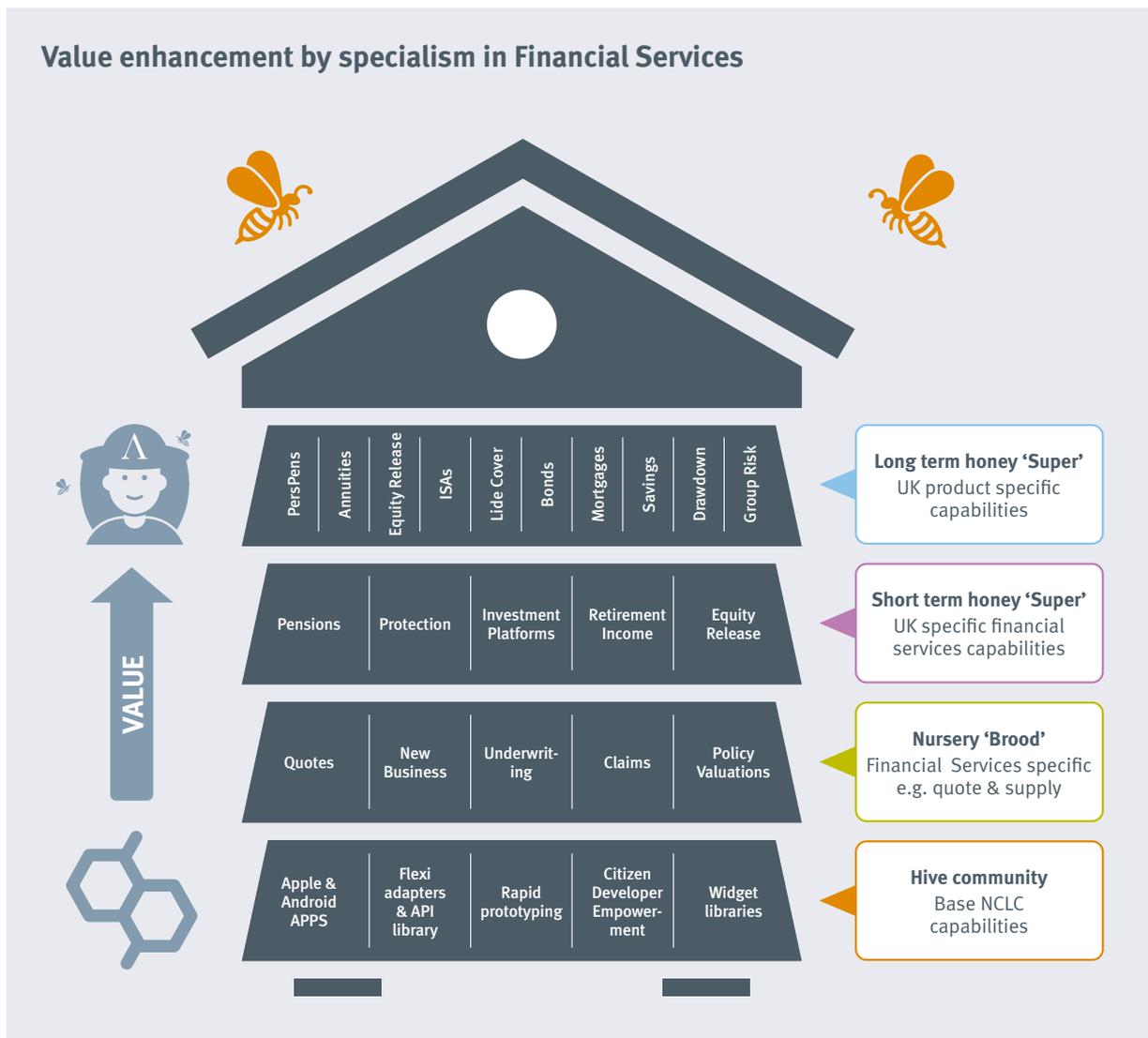

Figure 4: XX

Discussion Points (Cont.)

Important issues to consider in navigating a choice of NCLC vendor

As discussed in previous sections, in identifying a route through the LC technology stack, there are obstacles that need to be navigated.

- It is important to map a trajectory for the LC technology within your organisation. What benefits will the technology provide beyond the initial implementation, and will it continue to provide ROI? If there is no long-term strategy, can your organisation sustain the ongoing running costs of the technology?

- Will you be constrained by vendor lock-in? For example, are you coding in a platform specific language that cannot be ported to other systems? Can the code be managed in a way such that it is decoupled from the platform? It may be necessary to choose LC technology with a feasible exit strategy in mind.

- Coding, analytical, and logical design skills are still required in the technologies not considered completely NC. There is always a need for some level of technical skill or the willingness to upskill. It is not unusual to stumble across unforeseen technical challenges, for example tricky integrations with legacy systems, or complex domain model mappings. Therefore, it is wise to assume some level of complexity, and be able to fall back on trusted architectural patterns that require input from skilled individuals.

- Delivery of LC solutions and applications will still be part of an organisation's change program and subject to appropriate governance, just that the delivery is quicker. Traditional waterfall change processes may find it difficult to absorb such rapid delivery and it has already been noted that LC application delivery lends itself to the Agile methodology. For some this may be a significant shift in working practice, therefore an organisation needs to embrace new working practices at all levels.

- Ultimately LC technology delivers applications for business use cases. Therefore, maybe an organisation can protect itself from the risks previously discussed and utilise a 3rd party to deliver and maintain the LC applications. The benefits of rapid delivery and high frequency of change would still be achieved. However, this would have to be balanced with the risk of accumulating technical debt and making it difficult to change from one technology to another.

- Technology advances and trends move fast. In which case it would be fair to ask whether the LC vendor also moves fast. How quickly do they deliver updates or connectors to new systems? Or can the platform tools themselves provide the ability for the customer to quickly adapt and integrate to new external factors. For example, can the LC platform provide pre-built features that keep pace with advances in Artificial Intelligence (AI)? Many AI bots have learnt skills specific to Financial Services e.g. understanding and applying medical terminology in L&P underwriting and claims user journeys. Can the vendor integrate with them or do they have a partnership ecosystem making it easier for the Financial Services provider to acquire pre trained AI bots to enhance the value from the system they want to deploy using the NCLC platform? Or, is it flexible enough in being able to interface with new systems as they emerge? An assessment of synergy, and what technology best binds your existing enterprise applications together may be a high priority.

Roles within the Financial Service Providers 'colony'

Queen: As with a bee colony, there can only be one Queen bee or Programme Director / Sponsor. Clear direction is necessary to ensure scope is clearly defined, budgets monitored and the project delivers its objectives.

Workers: like worker bees, Citizen Developers are highly specialised subject matter experts (SMEs). There may be a lot of them in the financial service provider's organisation but they should have a clear understanding of their role and what they need to do on a daily basis in an agile development process for the benefit of the company / colony.

Grubs: whilst this might be stretching the analogy a bit, grubs are tomorrows' colony. We can think of them as SMEs who haven't yet become Citizen Developers. They may well be experts seconded from parts of the colony / provider, who show a keen interest in how the system develops and want to be part of shaping its' future. Whilst they might not be fed with Royal Jelly, they should be nurtured and encouraged in the same way.

Drones: like Drone bees, contractors or PSCs are specialists that tend to work closely with the Queen but have short lived lives in the colony and leave once their jobs are completed. They may be specialist data scientists, testers, graphic designers or other roles needed on a temporary basis.

Swarming: is when a successful colony replicates to create a duplicate. In a Financial Services provider this is similar to part of the organisation seeing the success of a project in one area or line of business as they start to use NCLC for the first time, then duplicating the successful development model, governance structures and NCLCP license to repeat the success in another part of the provider.

The Super Colony: Beehives don't exist on their own, they're always in groups. The grouping of hives can be compared to an FS provider using a separate hive for an LCAP such as Salesforce, another hive using FinTechOS for Life Insurance policy management, and another for robotic process automation handled by the Xceptor platform. Often, there will be an empty 'spare' next to a group with baited honeycombs so that when one of the colonies swarms it can go into a fresh hive to make it their home. Potential for a new NC platform to be incorporated into the colony, and a LC trained workforce ready to colonise.
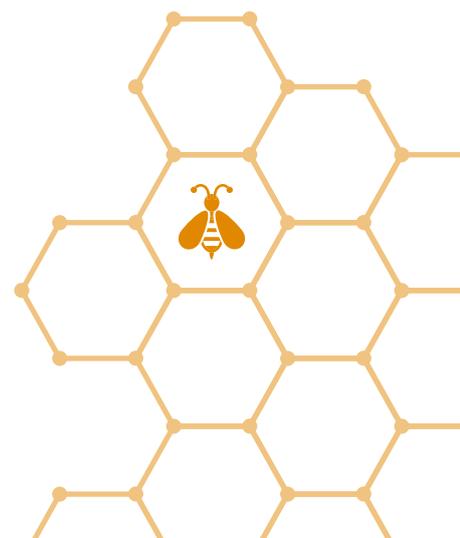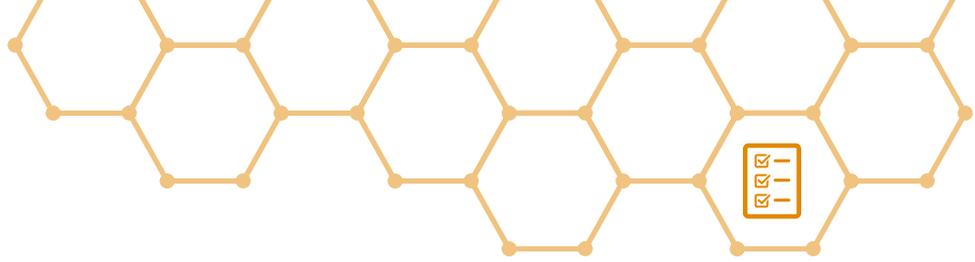
Conclusion

Digital capability will be essential to succeed in a 21st Century Financial Services sector and LC technologies will play an important role in delivering that capability. We see four key benefits to adoption:

- Applications can be built quickly enabling prototypes to be refined in iterations in a "test and learn" approach.

- Empowering Citizen Developers can shrink the gap between business and technology stakeholders helping to drive valuable innovation.

- Integration with external systems can be simplified through pre-built connectors.

- By selecting NCLC platforms that have evolved specifically for Financial Services, firms can benefit from pre-built and tested services.

This paper has summarised the range of LC solutions available, from platforms that specialise as "No Code", to powerful generic LC Application Platforms (LCAP), through to specialist solutions tailored specifically for the Financial Services sector.

Selecting the right one for your business will depend on your specific challenges. The need to deliver web applications for generic portals and mobile applications may point to selection of an LCAP. In contrast, selecting a Financial Services specific platform may be more suitable for replacing a legacy Policy Administration System. The message is to pick appropriately for the business problems being addressed and to sustain long-term value.

What is clear from the case studies is that those providers that are brave enough to take on the challenge can achieve some fantastic results in significantly shorter periods of time and for a lot less budget than would have been expected with traditional software development. We encourage you to reach out if you would like help planning a NCLC project and selecting the most appropriate provider.

References

– DunstanThomas. (n.d.). Dunstan Thomas Discovery Success Story. Retrieved from https://www.dthomas.co.uk/content/fs/clients/clients_discovery.shtml

– FinTechOS. (n.d.). Retrieved from https://fintechos.com/

– FinTechOS. (n.d.).

– Gartner. (2018). 5 Strategic Approaches to Digital Optimization and Transformation in Insurance. Retrieved from https://www.gartner.com/en/documents/3894472/5-strategic-approaches-to-digital-optimization-and-trans

– https://discover.xceptor.com/tei-casestudy-reg. (n.d.). Retrieved from Xceptor.com.

– Instanda. (n.d.). Instanda Case Study: Top 5 Global Insurer. Retrieved from https://instanda.com/case-study/top-5-global-insurer/

– Mendix. (n.d.). Mendix Customer Stories: Zurich. Retrieved from https://www.mendix.com/customer-stories/zurich-insurance/

– NoCodeJournal. (n.d.). State of NoCode. Retrieved from https://www.nocodejournal.com/state-of-nocode

– Xceptor. (n.d.). Retrieved from https://discover.xceptor.com/tei-casestudy-reg

– Xceptor. (n.d.). Xceptor Client Stories: Miller. Retrieved from https://www.xceptor.com/why-xceptor/client-stories/miller-insurance/

Rob Green
Consultant, Altus
✉ robert.green@altus.co.uk
in Robert Green

Will Watling
Life & Pensions Director, Altus
✉ will.watling@altus.co.uk
🐦 @WillAltus
in William Watling

Altus White Papers

With our focus firmly on the regulatory, strategic, propositional, operational and technological challenges our clients face, Altus understands the most pressing issues for financial services. We publish market insight, industry commentary and are at the forefront of industry debate.

ARTIFICIAL INTELLIGENCE

The evolution of financial advice

'Artificial intelligence: the evolution of financial advice' considers how computers can now 'outsmart the humans', and whether Artificial Intelligence can really deliver financial advice.

AUTOMATING ADVICE

Rise of the Machines

'Rise of the Machines' explores the theme of robo-advice and looks past current implementations to the future of a smart-connected, personal financial management, well-being and lifestyle digital adviser.
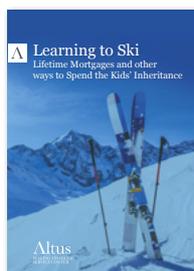
LIFETIME MORTGAGES

Learning to Ski

This white paper sets out to quantify the scale of the savings gap for baby boomers and how equity release will inevitably have to be used to help make up the shortfall. We then describe some of the challenges the industry must solve to make equity release more affordable for customers and as sustainable as possible for providers.

ADDRESSING COSTS

Giving your platform wings

'Giving your platform wings' is an uncompromising look at the challenge facing investment platforms as they continue to seek sustained profitability.

DISSECTING INDUSTRY ISSUES

The Lose-Lose Game

'The Lose-Lose Game' pulls apart the true cost of pension saving for vulnerable workers and questions the foresight of government policy.
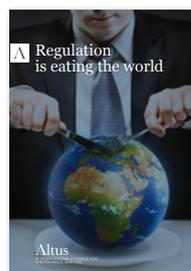
REGTECH

Regulation is eating the world

'Regulation is eating the world' examines how firms currently deal with the growing volume of regulation, some of the challenges involved and how new technology might help.

Queen Square House
Queen Square Place
Bath
BA1 2LL

+44 (0)1225 438 000
altus.co.uk

Altus

MAKING FINANCIAL
SERVICES FASTER