Reducing development and maintenance costs of web and mobile applications
MANAGEMENT SUMMARY

The Internet has had a profound effect on the way in which individuals live and work and businesses have had to dramatically change the ways in which they interact with their staff, customers, partners and suppliers to remain viable and competitive. The web plays an essential role in conducting mission critical business, and browser based solutions have been the norm for how end users interface with many types of business applications. This has resulted in a massive amount of application development on web-enabled front ends as businesses have looked to realise the true potential of the online channel.

However, with the recent explosion of web and mobile technologies, coupled with unprecedented high levels of consumer adoption of devices, the delivery of business applications has never been more complex. Consumers are now in control and are choosing the ways in which they wish to interact with businesses and are demanding optimum user experiences across all customer touch points. The complexities associated with delivering business applications that provide rich, consistent user experiences across a range of channels have increased enormously, and, with new channels and device types continually emerging, it is getting more complicated by the day.

For large enterprises such as banks and insurance companies with multiple product portfolios, the development effort is enormous. All too frequently business are being forced to make functionality or strategy compromises due to the development and resource costs associated with maintaining multiple web and mobile applications and front-ends. This is ultimately resulting in many web and mobile applications being built in silos across the organisation using disconnected and disparate tools.

The cost of maintaining these silo channels is further compromised by the reduction in productivity that occurs when traditional software developers are tasked with creating these applications. Despite the latest trends in development methodologies such as agile, there continues to be a challenge in alignment between the business and IT when determining business requirements for applications. This results in business needs not being satisfied and additional costs being incurred for application changes. There is a further cost implication as speed to market is impacted when applications are not fit to go live as they do not address what the business actually needs. edgeConnect, Temenos' User Experience Platform (UXP), addresses this problem by allowing business and IT to collaborate closely on the development, deployment, and management of web applications. edgeConnect also replaces the point-to-point chaos of existing infrastructures with a centralized architecture which, by separating business logic from presentation logic and the underlying data, allows the same application to be deployed over many different environments, with very little additional effort.

Furthermore, the move to mobile affects businesses in multiple ways, how they interact with their customers, with other businesses, and how staff work internally. Apps tend to have a shorter lifespan than traditional applications, especially for attracting consumers to a business’s services. The apps are the shop window to the world, and there is good reason why shops keep refreshing shop windows. To maximize productivity and reduce cost of development the capability to reuse common business processes, referred to as components, is a powerful benefit in a development environment; businesses must not build app silos. App sophistication is leading to context awareness resulting in personalized application; to manage the complexity of this new generation of apps it is necessary to first build in reuse.

For these reasons edgeConnect can significantly reduce the development time of building web-enabled applications, when compared to an alternative tools-based approach. Simple ROI analysis of development
using edgeConnect versus a purely frameworks tools centric approach shows that edgeConnect can yield 60% savings in development costs on a first project, rising to 95% on subsequent projects. This is principally due to the way edgeConnect manages the complexity of multiple channels, brands, users, and environments, such that costs do not spiral out of control. In short, given the way edgeConnect works, the more the business uses it, the greater the savings that should be realized.

This report identifies the business and technology drivers that are shaping the latest web and mobile applications, and illustrates the positive cost benefits and return on investment (ROI) that can be achieved with the edgeConnect development platform.

**BUSINESS DRIVERS**

The steady revolutionary action of the internet since its appearance in the 1990s is now accelerating. Following the web and web 2.0, mobility is now pushing the boundaries of the connected world even faster. This rapid growth of connectedness will not stop; on the horizon and already visible is the internet of things (IOT) where connected objects will communicate with each other without a human intermediary. For businesses this is both threatening and welcome: while change introduces uncertainty, it also creates new business opportunities.

The following business drivers help understand how these changes impact the customer and its relationship with businesses, and will allow businesses to flourish in this new world.

- **Customers in control of the channel:** Companies have traditionally been in control of the channels that link them to customers, from bricks and mortar, telephony, newspaper and television adverts, to more recently online web sites. The onset of smart technologies, spanning phone, TV, cars, glasses and more, with IOT increasing the range of connected objects, mean customers are no longer tied to a few controlled channels. The increasing use of mobile apps, whether native or web, is also threatening the time spent on the web. The implication for app and application development is that this broad range of channels cannot be cost effectively covered using traditional approaches; there are too many types of devices to support.

- **Online is the primary channel to customers:** A sound strategy is to create an online experience through which customers can be served, including mobile apps. The cost of being online has fallen dramatically for consumers and we are seeing a marked shift towards online activities versus traditional channels, as the richness of the online experience expands. Customers now perform more activities online, whether researching, purchasing services and products, socializing or collaborating and this trend will continue. The issues of availability, performance, scalability, security, and reach across all devices and networks (dynamically modulated for the available bandwidth) are all important hygiene factors for building a trusted brand online. Companies will need an application development solution that goes beyond frameworks and heavy amounts of custom code.

- **Mobile and further evolution of consumer behaviour:** The growth of mobile means that the shift to online presence is now affected by another consumer behavioural change: increasing use of smart mobile devices. Mobile app usage is quite varied and often downloaded on demand: apps that are unconnected, apps whose functionality is based on being connected, and apps that filter the web for smart device consumption. We can expect consumer behaviour to continue to evolve with the onset of IOT, as well as further growth of smart mobile device usage in the home and office. Companies will need to be able to monitor and refine customer experiences using behavioral monitoring and analytics, as well as support traditional marketing techniques like A/B split and multivariate testing.
They will need the ability to make changes quickly, while not breaking good governance, in order to remain fresh and engage with users: this need will grow in importance.

- **Evolved application development**: There is a need for development speed due to the speed of change in user behavior. Traditional writing of software code suffers from the miscommunication gap between business and IT. Given the business drivers identified above there is a need to be able to respond swiftly to market changes, and create rich user experiences that are consistent and reflect well on a brand across a heterogeneous consumer device landscape. One answer to this is end-user development, by which is meant application development produced by business domain experts.

- **Creating a smarter user experience**: Creating silo-built channels, i.e. internet banking, mobile banking, introduces complexity in managing all the different products and services, for all the brands and user types, and is therefore also more costly. A better approach is context-aware computing that can take into account all of these different consumption methods within a single context-aware application. These contexts include location, bandwidth availability, time, skills, and even emotion that will drive applications to be richer in their dynamic behaviour such that every user has a unique experience created specifically for their circumstance. Application development that can exploit such context will have an advantage over traditional approaches for many business applications.

We are seeing organizations struggle with the costs and issues of application development and deployment, particularly when looking to exploit the reach and range of web and mobile web platforms. Making changes to a process may spawn a range of dedicated development projects as each application in turn is modernized. Not only must the business consider how it manages the complexity of its channels, products, brands, customers, etc. but it also needs to give due care and attention to the interface it uses when interacting with the different individuals. Of course different types and groups of users have different needs and privileges, and different types of product or service may be better communicated via certain mechanisms than others. This needs to be considered as a business decision, not one dictated by technology.

We need to move web-based application development and management to more of an adaptive, business-focused level – one that is supported by software that minimizes development costs while allowing the organization to retain control of key processes over a number of different channels.

**TECHNOLOGY IMPACT**

The following technology drivers help understand how software development is changing:

- **Development speed**: A good metric for the effectiveness of technology is time-to-market – how quickly can the application be assembled and delivered according to the original specification and requirements. The sooner an application can be deployed, the sooner the business can start using it as a channel for saving costs and/or driving additional revenue. However, a problem with purely focusing on time-to-market is that the ongoing costs of maintenance tend to get overlooked. Maintenance dominates the total cost of an application lifecycle; it consists of defect fixing but more so functional enhancement and extension. The key consideration when making an assessment between application development options is how easily changes can be made not just during creation but also in production: the comparison of development options therefore needs to account for the long term cost of ownership.
• Governance: At the same time that development speed is required we do not want to increase risk. This is where governance plays an important role and a platform solution is able to monitor development activity to ensure compliance with company standards and regulations.

• Simplification: As businesses implement their web and mobile strategies, using the reach and power of the web to engage with as many customers, clients, and other stakeholders as possible, they end up using a mix of web/mobile presentation mechanisms, depending on the channel, product, brand, and audience. Change requests become a major technological challenge, having to manage multiple versions and technologies, and inhibit the process of change. What appears to be a simple change request at a process level can ripple through to a large number of different applications, with the same change needing to be made in each environment. This has a knock on effect on the business, which needs to be responsive to market opportunities and be able to deliver changes quickly. Therefore a single tool for all devices/platforms/browsers rather than several is able to implement changes once and publish the changes to multiple end points.

• Framework soup: For web and mobile applications that need to reach a consumer audience across a wide range of devices available in the market today, the development options tend to fall into two main categories: HTML5 frameworks and libraries (framework soup) versus platforms. The underlying software architectural choices between these options play a pivotal role. The combination of frameworks and libraries can result in an architecture that lacks integrity, having quite a complex multiplicity of hooks between the various packages used, as well as issues of security consistency across the components, performance consistency across the components, and cross-browser support. Many open source frameworks are transitory as developers move onto the ‘next best thing’ and for a business these transitions escalate costs. A few years ago Java frameworks Struts and Spring and .Net MVC framework were popular, now AngularJS Ember.js and other JavaScript frameworks have risen in popularity, and these in turn will likely be superseded in a few years’ time.

• Platform approach: In contrast to a framework soup, a single platform that is based on a modular and layered architecture has the benefit of integrity and simplicity in-built from the ground up; a platform solution is pre-tested for security, scalability, and performance. Organizations that try to build their own user experience platform out of a framework soup incur many hidden costs. The next section describes edgeConnect UXP.

EDGECONNECT USER EXPERIENCE PLATFORM (UXP)

dedgeConnect was principally designed to create a development environment that non-technical business analysts and non-programming domain experts could use intuitively, and build the presentation layer of applications rapidly, and republish to multiple devices without any additional work. The product comprises an integrated development environment (IDE), edgeConnect IDE, and a runtime environment, edgeConnect RTE. Thus users of edgeConnect are mainly business domain experts who create the application UI based on their deep understanding of the application requirements. Organizations that need to rapidly ramp up a project, for example to exploit a new market opportunity, will find it easier to do so with edgeConnect as they can rely on business domain experts, whereas its IT department may be tied up on large scale projects and have few resources to release within a short notice period.

dedgeConnect combines an easy to use development environment for creating prototypes with an evolutionary prototyping methodology, or early visualization approach (EVA). edgeConnect manages the architecture of the solution thereby allowing prototypes to naturally evolve to the final business solution. The solution also implements the model view controller (MVC) pattern for building web applications, where the presentation
layer (view) is separate from the database and business logic (model), with the controller performing the task of interpreting user interaction and coordinating traffic between the backend and the presentation layer. This also allows integration points to be clearly identified, and abstracted from the other parts of the application. This separation of activities means that changes in one layer, such as presentation, can be made without affecting the other layers, speeding up maintenance tasks.

Thus edgeConnect automatically constructs applications with a layered architecture, and this approach means that a single model can be associated with multiple presentations. Starting with a single application process definition, these multiple presentations may be generated to support:

- **Brands**: for white/private labeling.
- **Channels**: Call center, internet, and mobile.
- **Users**: Customer, staff, intermediary, and partner.
- **Languages**: Product internationalization. Language may also vary for channels e.g. mobile will require short text and call center users will require verbose call scripts.
- **Devices**: PC, smartphones, tablets, surface tablets.

An application workflow process is also separated from integration points within edgeConnect. Process rules are used to action integration points so a process knows when to perform an integration task. The process does not have any details of the integration being performed, therefore a process will not be affected if there are changes to the integration mappings, for example if a table in a database changed, or even if the database changed, or the integration changed from database to web services. Where multiple presentations are created from a single process definition, changes to that process will be automatically propagated to every presentation. If entire processes, process steps, or process fragments have been re-used in other processes, changes are propagated to all re-used instances and all presentations of those processes. These levels of reuse and automated change propagation have a dramatic impact on productivity.

The target deployment models may be online (rich or thin clients), offline, portal, and composite. It is possible to insert scripts into the final output, and tie in other applications through say Java or web services. There is no need to manually change the generated code. edgeConnect’s point and click GUI based approach does away with scripting for common operations such as field validation, and local calculations. More advanced GUI features such as calendar controls and spell checking are also provided. The solution enables non-technical business users to design and build applications. It allows marketing staff to apply look and feel and branding independently of the logical business solution. It also allows IT to work on complex integration tasks. The three parties can work within the same environment simultaneously. edgeConnect provides change management and tracking capabilities with its team-based Enterprise release of the product. For business users edgeConnect’s zero code approach enables them to focus on functionality rather than programming syntax, and promotes a much closer involvement in the application lifecycle.

**ROI ANALYSIS**

ROI analysis of software development tends to focus on the development phase, and speed of development is important, but there is another more crucial phase and one in which most development approaches have little to offer. Maintenance is the largest expense in the total lifecycle cost of an application and it is not just about fixing defects as the largest maintenance budget is spent on changing and enhancing functionality. Some estimates put the maintenance bill as high as 80% of the total lifecycle cost. So an environment that reduces the maintenance burden has a significant impact on the total cost of an application.
To understand the qualitative ROI benefits of edgeConnect, Ovum has produced a model based on three comparison criteria: lifecycle costs, new project resource savings, and existing project in production with a change request and the reuse savings. Table 1 shows the model and the ROI benefits along the complete lifecycle, from requirements to maintenance.

The most significant feature of the edgeConnect technology is that it simultaneously targets each of the main lifecycle cost issues, leading to potentially significant savings, both in terms of initial development and on-going maintenance and management. Ovum’s model provides a qualitative example with quantitative savings estimates when using framework tools and libraries versus edgeConnect. These estimates, though indicative of the potential savings, are backed and verified by Temenos as having been borne out by edgeConnect users. Clearly deployment scenarios are far more complex and individual than can be captured in a simple ROI model, but this analysis provides a useful starting point.

### Table 1: Application lifecycle costs and edgeConnect benefits

<table>
<thead>
<tr>
<th></th>
<th>Traditional cost issues</th>
<th>edgeConnect benefits</th>
<th>Resource required, % of total</th>
<th>New project edgeConnect % savings</th>
<th>Reuse with edgeConnect % savings</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The phases of the application lifecycle from inception to production.</strong></td>
<td>The challenges encountered using a mix of HTML5 frameworks and libraries.</td>
<td>How edgeConnect’s platform approach addresses each challenge encountered in a framework approach.</td>
<td>Typical resource spread in a new project.</td>
<td>How edgeConnect brings savings to this resource spread.</td>
<td>Reusing of existing application with small change request.</td>
<td>The rationale for savings.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Poor requirements capture.</td>
<td>Business domain experts engaged.</td>
<td>2</td>
<td>0</td>
<td>90</td>
<td>Similar requirements.</td>
</tr>
<tr>
<td>Design</td>
<td>Up front heavy design.</td>
<td>Rapid and iterative prototyping.</td>
<td>2</td>
<td>50</td>
<td>90</td>
<td>Rapid prototyping of new UI.</td>
</tr>
<tr>
<td>Development</td>
<td>Manual coding.</td>
<td>Auto-generated code.</td>
<td>9</td>
<td>60</td>
<td>95</td>
<td>Only make a few changes.</td>
</tr>
<tr>
<td>Testing</td>
<td>Testing of manual code.</td>
<td>Auto-generated code reduces defects.</td>
<td>6</td>
<td>25</td>
<td>70</td>
<td>Test the changes and run regression tests.</td>
</tr>
<tr>
<td>Release</td>
<td>Multiple deployments and runtimes. Extensive re-work.</td>
<td>Single model generates multiple presentation releases.</td>
<td>1</td>
<td>30</td>
<td>50</td>
<td>Automated deployment, some local admin.</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Time consuming and expensive. Duplication of effort for changes across multiple deployments.</td>
<td>Easily make changes and see these propagate across all releases.</td>
<td>80</td>
<td>50</td>
<td>100</td>
<td>This exercise is a maintenance change request.</td>
</tr>
</tbody>
</table>

Source: Ovum
edgeConnect has a host of development productivity features to expedite a project process, using integration components, automated project documentation, a zero-coding environment that uses wizards, drag-and-drop (within and across editors), and automatic UI generation. In comparing frameworks versus a platform solution (like edgeConnect) the ease of making changes should be examined. For example, edgeConnect will automatically create a UI field to reflect a change in the underlying data model with just a simple update of the linked web service.

The Ovum model does not account for additional savings that could be made by edgeConnect’s auto project generation capability. An organisation with existing databases, XML schema or web services can generate entire projects (screens, validation, and integration) from these assets. Hence savings for initial projects could still be greater depending on what assets are available to edgeConnect.

ARISTA INSURANCE CASE STUDY

BACKGROUND TO IT DEVELOPMENT AT ARISTA

Arista Insurance is a commercial insurance provider specialising in property, liability and motor insurance protection for businesses throughout the UK. Dedicated to providing insurance through its network of regional brokers, Arista does not deal direct. This business model demands that Arista provides the very highest levels of service to its channel, encouraging loyalty and enthusiasm from its broker partner base.

Arista launched with a small team in 2007 and has grown rapidly to 110 staff operating out of seven offices and is currently achieving Gross Written Premiums in excess of £80 million.

Arista’s business model meant that it needed to provide from the outset the very highest levels of service to its broker channel and the development of a web-based portal was a critical element of this to enable brokers to trade online. Arista chose edgeConnect for the development of this business critical broker portal.

Arista took the decision from its outset not to have a traditional IT department but to outsource IT infrastructure and functional development while, however, retaining in-house all of its product development capability. Its business systems group is nine people strong, of which two are active developers on edgeConnect technology for insurance products, and the others have a range of skills from project management to business analysis and support functions, with backend functional development and hardware management done by the outsourcer.

As a newcomer to the commercial insurance industry Arista has marked itself out by being innovative in its use of IT to deliver speed to market of new products and to enable an exceptional quality of services to its brokers. This has been recognized within the insurance industry; it has been declared a winner for four awards: 2013 Insurance Times Awards, MGA of the Year 2013 and 2012 Underwriting Service Awards, Commercial Combined Team of the Year; and 2007 Insurance Times Awards, New Entrant of the Year.

THE CHALLENGE WAS TO CREATE A SELF-SERVICE WEB PRESENCE

Arista’s business model was for its brokers to use a self-service facility accessible from its web site with full integration to its back-end systems. Arista set itself a challenging goal of launching five products to market and establishing in house development capabilities within a 12 month timeframe. Speed to market was absolutely critical to the success of Arista.
The founders of Arista had evaluated edgeConnect in a previous organisation and knew that it was the technology they needed to enable them to achieve both their initial goals and support the business model moving forward. The first edgeConnect project kicked off in August 2007. Arista had a working pilot that went live to a small panel of brokers in December 2007. It was then fully released to all appointed brokers in January 2008: This was a fully integrated web based portal with full integration to the back office systems, all developed in less than four months. The project comprised three teams: the Arista team with two developers, a team of three people with edgeConnect skills from edge IPK, (later acquired by Temenos), and the outsourcer variously had five or six people creating mainly web services to enable integration to the back office systems. The Arista developers went on an initial three day edgeConnect training course and then immediately started working on the first couple of products alongside the edgeConnect skilled team. After the third product was launched to market Arista became totally self-sufficient in terms of its in-house development using edgeConnect and went on to develop and launch seven further products in less than 18 months.

edgeConnect proved immediately to be a good cultural fit with Arista’s business model of outsourcing its IT infrastructure while keeping web technology development in house delivered by its business domain experts. The edgeConnect environment enables Arista’s business analysts to develop applications and focus on the business requirements. Furthermore edgeConnect adopts an agile-like, rapid development approach which continues to be very important to Arista as it not only enables speed to market of new products but also enables any product or presentation changes to be made easily and quickly, reducing maintenance costs and allowing Arista to be highly responsive to customer requirements and changing market conditions.

ROI IN USING EDGECONNECT AT ARISTA

The launch of the web based broker portal has enabled Arista to provide an industry recognised high quality of service to its brokers, and by using edgeConnect it has also achieved significant ROI savings. The ROI is a result of improvements in productivity around product development and speed to market and also operational savings.

Traditionally insurance IT teams develop complex specifications that attempt to capture functional product and underwriting requirements from various business stake holders. IT teams then attempt to build business and product applications piece by piece. Turnaround times are typically in excess of nine months. With Arista business analysts using the agile development approach of edgeConnect, Arista is able to develop fully working applications for new products in around 8 weeks. This results in products getting to market in at most a third of the time needed with other approaches.

Dave Cheeseman, Head of Business Systems at Arista, draws comparison between his previous company where a similar product line required a team of 27 people and Arista which requires only 8 people – this is equivalent to a 70% staff saving. Staff costs are among the highest factors in IT expenditure so savings here provide a significant ROI benefit.

The web based portal has also enabled Arista to streamline many business processes with time saving automation and allow staff to focus on business generation rather than administration tasks. The system also provides quotes for all the cases that arise, unlike some online insurance systems that cherry pick the easy to process applicants and refer more complex ones for manual processing. Arista estimates that it has saved £2.1m over a five year period through operational cost savings that the automated process introduced. Furthermore, Arista state that their decision to use edgeConnect to develop their web based portal has significantly contributed towards Arista pushing through £80m of written premiums.

TEMENOS CASE STUDY
BACKGROUND ON TEMENOS AND THE EDGE IPK ACQUISITION

Temenos is a global provider of software systems to financial services companies and edge IPK was an independent software vendor with its flagship product, edgeConnect, a User Experience Platform (UXP) for developing web based, transactional applications. Temenos and edge IPK had been working in partnership as users of various Temenos core banking platforms were looking for an alternative front end to the existing Temenos offering. The Temenos customers wanted more flexibility and control of their user interfaces, the ability to integrate to multiple host systems and be able to deliver optimum user experiences across all channels i.e. web, mobile, call center – the existing Temenos Channels solutions were tightly coupled with the back end core systems and did not provide this level of flexibility. It soon became apparent that edgeConnect was the way forward. Temenos recognized the power of the edgeConnect platform and the multitude of opportunities that it gave Temenos in being able to completely ‘revamp and modernize’ not only its Channels solutions offerings but also the internal user interfaces of its core banking platforms. Temenos decided that it would acquire edge IPK.

edge IPK had invested over 13 years of research and development in edgeConnect with a pure focus on the ‘presentation layer’ or ‘User Interface’ and had a solid roadmap in place to embrace continually emerging technologies and subsequent consumer expectations and demands. Temenos wanted to leverage the opportunities this presented and decided to not only use edgeConnect to enhance its own channel offerings such as internet and mobile banking but to continue to sell edgeConnect as a standalone User Experience Platform, to non-existing Temenos Customers within financial services and across other vertical sectors via its partner network. Guaranteeing the independence of edgeConnect ensures it meets market needs and this serves the long term interests of Temenos as its products remain at the cutting edge.

In addition to core banking platforms Temenos also has a native mobile banking product. The various user interfaces used within the core banking and mobile products resulted in about 12 UI sets built on different technologies, different vintages, and all getting out of date and unable to meet the current demands of the customers. For example rendering an HTML page in 8 seconds was acceptable a few years ago for an internet banking application, whereas today it needs to load in less than 2 seconds. There were a few projects in progress to replace some of these UI sets, one was a web design from scratch using Apache Wicket, a web application framework, and another project for updating T24 Bank Staff was based on Adobe Flex. The latter was ready just as Adobe dropped Flex in favor of HTML5. In the case of Wicket the time scale to deliver the product was too long, the development skills needed were scarce, and the capabilities of the deliverable had deficiencies. It was at this point Temenos investigated the market for user experience (UX) platforms, discovered edgeConnect, first partnered with edge IPK and then bought it outright.

Temenos has at time of writing had 18 months experience working with edgeConnect, at first in the partnership with one customer and then as an internal business unit. The applications needing modernization had to deal with moving targets. The internet banking products prior to edgeConnect were not flexible enough to meet customer needs. For example one customer wanted to use a third party design agency to create the UI and implement that in the solution and the Temenos system could not allow that. So even quite small banks today want a high degree of customization of the user experience, want better usability, greater functionality, and particularly performance: one Temenos customer is demanding web page load times within 1.5 seconds. Meeting such performance targets would not have been possible with the previous version of the products. With such changing requirements the comparison of pre and post use of edgeConnect needs to take into account how edgeConnect is able to stretch the boundary of non-functional requirements.

EXAMPLE BANKING APPLICATION PROJECT
In a typical internet banking project the staff used comprises a UK based business product manager and a technical product manager, and in India Temenos has developers building the product, both the UI and integration, a UK based design authority working with an agency, and testers. There are a couple of highly skilled edgeConnect developers who produce proof of concept designs to guide the other developers. The majority of the developers are in a team which a year ago was working on a completely different product, so they have been switched to edgeConnect in rapid time.

In one particular internet banking project, six Wicket developers engaged on the UI for about nine months before the project was canned in favor of edgeConnect. In that period the project had progressed to about 30% of completion. With edgeConnect the whole project was completed in less than six months.

Table 2: Example internet banking project comparison

<table>
<thead>
<tr>
<th></th>
<th>No. of developers</th>
<th>Cost of developers</th>
<th>Time to completion</th>
<th>Training overhead</th>
<th>Staff retention</th>
<th>Application performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>JavaScript frameworks</td>
<td>6</td>
<td>£300k pa</td>
<td>36 months</td>
<td>High</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>edgeConnect</td>
<td>6</td>
<td>£120k pa</td>
<td>&lt; 6 months</td>
<td>Low</td>
<td>Good</td>
<td>Superior</td>
</tr>
</tbody>
</table>

Source: Temenos

In another use case example, a modernization/migration project, Temenos is replacing the UI in its core banking product T24, and there are tens of thousands of screens involved. Manually changing these screens is just not feasible as you would need a huge number of developers and there would likely be inconsistencies because of the scale of the undertaking. So Temenos decided it needed a high degree of automation in screen production and then manually tweak the more important screens. Here highly skilled edgeConnect and Java developers worked on producing the automation. These automation tools were used once to change the existing product and would then also be used by clients to convert their existing UI screens. This project also identified a number of enhancement opportunities to edgeConnect itself, so benefits to all edgeConnect users ensued. Most of the project work was in the automation tooling, which generates XML automatically, rather than having developers use screen editors to manually change the UIs. Again, the use of edgeConnect made dramatic improvements in productivity given the scale of the undertaking.

DEVELOPMENT OVERHEADS

In using HTML5 frameworks there is a combination of different tools that need to be used, so in addition to Wicket there would be widgets, and security frameworks. Most projects end up using a unique combination of frameworks which leads to compatibility problems. There is also the problem of having to find skilled developers in all these different frameworks, so there is significant training overhead. In comparison edgeConnect is a single product that covers cross-browser compatibility, multi-lingual internationalization, multi-form devices from smart mobile to tablet to PC, and elements of security. Productivity is therefore greatly improved. For testing projects built in edgeConnect the modularization saves considerable time due to the high degree of component re-use, which means testing only needs to be performed on core elements. This modularization also entails a high degree of consistency of look and feel in the product. Overall, every aspect of development is done quicker using edgeConnect.

edgeConnect has the useful feature that it allows a web service to be re-imported and it will not over-write the UI already done but it will spot the differences. So if the back-end has been enhanced and a new field appears in the data model then edgeConnect will automatically pick this up and render it in the front-end. So it has good tools for back-end enhancements.
The long term aim for Temenos is to use edgeConnect for all its product front-ends. This reduces the cost of building and maintaining its product portfolio, and also allows mixing and matching components across different products. So if a client purchases two Temenos products they can be provided with a single edgeConnect built UI to span both products. In essence Temenos is ‘practicing what it preaches to its clients’ with regard to having a single, consistent user experience platform across all systems.

SOFTWARE TESTING

The proprietary nature of Temenos products meant that pre-edgeConnect testing was done in-house. Post edgeConnect the internal testers were supplemented with a partner. It was considerably easier to bring in an external tester to work on an edgeConnect project, because the tool was easier to use than a multitude of frameworks, but there was also an incentive for the testing outsourcer. Investing in training its staff in edgeConnect meant that the partner had guarantee of future work using the tool rather than testing proprietary interfaces which require multiple skills.

The cost of testing has not changed much pre and post edgeConnect, internally Temenos used India based testers in both cases, and using the external contractors the cost per tester was higher but the testing was completed in quicker time. So costs were overall the same as higher costs per tester were offset by needing them for a shorter time.

THE VALUE OF MEETING MARKET NEEDS AND BEING FAST TO MARKET

Consumers are now dictating the ways in which they wish to interact with their financial providers – the customers are now in control and financial services companies are being forced to respond and change their business models. The quality and consistency of the user experience provided across all touch points coupled with the speed to market of new innovative products, are becoming the key differentiators between providers and critical in customer retention and acquisition.

Banks looking to buy or replace their core banking platform will also be taking into account the channel solutions available to them out of the box as consumers look to interact with their provider mainly through the digital channels rather than the more traditional channels such as the branch.

Temenos is now able to offer its customers best of breed solutions with its leading core banking platforms coupled with channels solutions underpinned by a future proof platform, edgeConnect. This enables speed to market for new products across multiple channels, a superior omni-channel user experience and the ability for consumers to access their banking services anytime, anywhere and via any device.

MAINTENANCE AND POST PRODUCTION BENEFITS

A key benefit of edgeConnect is that it has separate design and runtime environments. The entire configuration of an edgeConnect project is in XML, stored under its source control repository, and deployed onto the various runtime environments. Pre-edgeConnect Temenos did not have a runtime environment but was planning to build one so edgeConnect provided that advantage. Separating design and runtime is the key aspect, which offers the capability of merging updates.

Temenos provides a model bank solution which a client then modifies for its needs. Whenever Temenos subsequently updates its model bank the client would not be able to update its implementation unless it did manual merging, which is a time consuming process. The client would be able to update the core underlying system but it would not be able to take advantage of any of the new front-end functionality, unless done
manually. The metadata is separated in the edgeConnect modular system, meaning that merging can be automated, with manual resolution of clashes. This approach allows a bank to easily upgrade to the latest model bank release.

The day to day production administration pre and post edgeConnect is unchanged but maintenance changes are much faster and safer to implement. The reduction in manpower to implement upgrades is considerably reduced. A tier one bank would normally use four times as many staff to upgrade a system as a tier two. Using edgeConnect halves the total number of staff devoted to upgrades.

BUILDING MOBILE CLIENT INTERFACES

When edgeConnect was acquired it did not have a mobile capability. Temenos has enhanced the capability of edgeConnect by enabling it to support hybrid mobile applications on mobile and tablet devices. So the same tool can be used across any UI front-end. Temenos has also implemented responsive design, so the user experience across different form factors is ensured to be good. A product called Twitter Bootstrap is used with edgeConnect to give responsive design, with data formatted differently or have fields left out. Temenos now has responsive design on all its UI, but with the hybrid mobile, which is effectively a mobile app whose content is populated by HTML5. So this has the advantage of common tooling across web and mobile, but there are different dedicated apps for iOS, Android, and Windows for an optimum mobile experience, with a lot of attention given to mobile performance by using native capability. Thus the apps are built natively for each mobile smart device OS, such as the container and pre-login, login, and security functionality, and the content is populated by HTML5 provided by edgeConnect. This approach means the apps are quite versatile and can cope with different user roles.

CONCLUSION

As the majority of customer interactions move online and the sophistication of the user experience they demand increases, organisations must take a smarter approach to online product development to succeed. An approach that transforms speed of development, improves productivity, and future proofs against changes in technology is essential. This paper examines the advantages of using edgeConnect in terms of ROI. We have found significant savings across the lifecycle: during development and testing (estimated as 15% of the total lifecycle) edgeConnect can offer 85% savings. During maintenance, estimated as 80% of the lifecycle, edgeConnect can offer 50% savings, and thereafter 100% on reuse projects. The real world case studies reflect these significant cost savings: Temenos was able to reduce its development staff costs by 50% within a completion time that was 16% of the time it took traditional development. Similarly Arista was able to reduce its staff by 70% compared with an environment supporting virtually an identical product set.

We expect many more new digital channels and devices to emerge in the coming years and edgeConnect’s ‘write once, publish many times model’ will continue to offer benefits as users can continue to use their existing edgeConnect skills while their counterparts chasing the latest new technologies will be taking on costs in re-skilling and re-tooling. Temenos effectively removes these costs from users by taking on the evolution of edgeConnect to whatever next generation technologies appear. Today the mobile wave is having a massive effect on businesses as they look to have a presence on mobile devices. edgeConnect’s hybrid mobile support ensures users have all the options available to build a mobile apps strategy that suits their business and their customers.

Effective use of the web and mobile as channels for both external and internal applications, across all manner of devices, is of critical importance in conducting business for all organisations. However, many of these
businesses are finding that their ambitions and goals are being compromised and hindered by the sheer complexity of the IT architecture that they are using to try to execute their strategies.

The situation ought not to be surprising: mergers and acquisitions, tactical investments, changes to business processes, and the addition of new products, brands, channels and devices compound what is already a complex situation, leading almost inevitably to the formation of hard to modify application designs and disconnected pockets of IT. As a strategic enterprise platform Temenos edgeConnect helps these businesses unravel application complexity, replacing them with a modular architecture that allows business applications to be developed and deployed without any coding and for these applications to be repurposed, re-used or re-rendered for varying channels, brands, users, products, processes or geography, with significantly reduced effort.

edgeConnect places an emphasis on the ability to preserve the intellectual assets of the business in a form that is not constrained by impenetrable code, but which can easily be understood, communicated, evolved, and maintained. It allows the organisation to develop its software assets without the restriction of deployment to a single current environment, or concern about future waves of software architecture. Productivity features such as the ability to rapidly create application prototypes helps both business and IT roles work together on development projects using familiar concepts and terms, whilst communicating and integrating effectively.

### Table 2: Contact Details

<table>
<thead>
<tr>
<th>Temenos Headquarters SA</th>
<th>Temenos UK Ltd</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Rue de l’Ecole-de-Chimie</td>
<td>71 Fenchurch Street (5th Floor)</td>
</tr>
<tr>
<td>CH - 1205 Geneva</td>
<td>London</td>
</tr>
<tr>
<td>Switzerland</td>
<td>EC3M 4TD</td>
</tr>
<tr>
<td>Tel: +41 22 708 1150</td>
<td>United Kingdom</td>
</tr>
<tr>
<td>Fax: +41 22 708 1160</td>
<td>Tel: +44 (0) 20 7423 3700</td>
</tr>
<tr>
<td></td>
<td>Fax: +44 (0) 20 7423 3800</td>
</tr>
</tbody>
</table>

Source: Temenos

---

**APPENDIX**

**METHODOLOGY**

Ovum’s research was assisted by briefings with Dharmesh Mistry, co-founder and CTO of edge IPK, a Temenos company, Robert Burch, Product Director for Channel Banking, Temenos, and Dave Cheeseman, Head of Business Systems at Arista Insurance, and Ovum is thankful for their input.

**AUTHORS**

Michael Azoff, Principal Analyst, Software Solutions Team

 michael.azoff@ovum.com

Ian Charlesworth, Director of IT Research and Analysis, Ovum

 ian.charlesworth@ovum.com
OVUM CONSULTING

We hope that this analysis will help you make informed and imaginative business decisions. If you have further requirements, Ovum’s consulting team may be able to help you. For more information about Ovum’s consulting capabilities, please contact us directly at consulting@ovum.com.

DISCLAIMER

All Rights Reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the publisher, Ovum (an Informa business).

The facts of this report are believed to be correct at the time of publication but cannot be guaranteed. Please note that the findings, conclusions, and recommendations that Ovum delivers will be based on information gathered in good faith from both primary and secondary sources, whose accuracy we are not always in a position to guarantee. As such Ovum can accept no liability whatever for actions taken based on any information that may subsequently prove to be incorrect.