Bloor

InDetail

InDetail Paper by Bloor
Author **David Norfolk**
Publish date **January 2015**

Uniface for High Productivity Mobile Development

The prime use case for Mobile Development in Uniface is the addition of a mobile channel to existing Enterprise applications, as Uniface's model-driven development approach is able to re-use existing automated business logic and data structures.

Author **David Norfolk**

Executive summary

Uniface is an established high-productivity model-driven development platform which is particularly suited to adding a mobile channel to existing applications, because it can re-use existing business automation assets, both Uniface and non-Uniface.

2014 was the year when the use of the Internet from Mobile devices overtook its use from desktop systems.

Uniface agrees with analysts who think (see *http://searchenginewatch.com/article/2353616/Mobile-Now-Exceeds-PC-The-Biggest-Shift-Since-the-Internet-Began*, for example) that 2014 was the year when the use of the Internet from Mobile devices overtook its use from desktop systems (although a lot of people still probably choose the desktop for their most important or potentially risky transactions – it's easy to make mistakes on a small screen with small buttons). It wants to position Uniface to be the enterprise platform of choice for delivering mobile interfaces to Enterprise systems. Obviously, Uniface has strong competition in this role; but it also has competencies that make this a not-unreasonable ambition for 2015.

Uniface Mobile is based on the modernisation of the existing Uniface development platform, made possible by the funds made available by Marlin Equity Partners, which acquired Uniface in 2014. The new funds have allowed Uniface to bring forward functionality from Uniface 10, as a new upgrade to Uniface 9, version 9.7; although Uniface's full mobile development capabilities will still be delivered with Uniface 10. This will be targeted at new customers to start with, with Uniface 10 WE (Web Edition) as the first commercial release in Quarter 1 2015; the migration path from Uniface 9.6/9.7, for client/server customers, will come as a first point release; and Version 10 ME (Mobile Edition) will arrive in the 3rd quarter of 2015. Readers should reference Bloor's Uniface 9.6 whitepaper (*www.bloorresearch.com/research/indetail/compuware-uniface-9-6/*) for the general strengths of Uniface, which still underlie its Mobile capabilities in Version 9.7 (and, indeed, Version 10 ME).

The prime use case for Mobile Development in Uniface today, is the addition of a mobile channel to existing Enterprise applications, as Uniface's model-driven development approach is able to re-use existing automated business logic and data structures. It generates components with new mobile capabilities as part of a highly-productive process, built around Uniface's core values of productivity, vendor neutrality and reuse of existing assets.

None of this would matter if the Enterprise didn't want to develop a strong Mobile channel; but it does. Google sums up the reason for this succinctly in its marketing campaigns. Google says (in its Google Mobile Ads Marketing, Sept 2012) that visitors to a friendly website are 67% more likely to buy; whereas an unfriendly website makes them 61% more likely to leave. People don't buy smartphones without wanting to use them, so Enterprises are finding that their e-commerce sites are increasingly servicing Mobile customers – and they need a Mobile interface that attracts and keeps customers. An effective Mobile site is becoming a key part of building a strong brand and cementing long-term customer relationships.

An effective Mobile site is all about the User Experience: delivering convenience, productivity, utility, responsiveness, attractiveness – and never forget the need not to waste battery power. This accords well with the User Experience focus in the coming Uniface v10; and Uniface's recent takeover has given it the resources needed to build a compelling Mobile offering, in the short term (based on Uniface 9), as a poster child for the new, revitalised Uniface.

The Uniface Mobile offering will include new mobile development features and Web enhancements. Uniface now has a dedicated mobile development team, with new developers (guided by a Uniface "old hand"). Uniface's "unique selling points" for mobile development are its easy reuse of existing business logic and application data, coupled with Uniface's well-tested high productivity approach.

Uniface Mobile is the first fruit from a revitalised, newly-private Uniface. Uniface can provide the highly-productive environment and new technologies needed to make this successful.

Fast facts

Uniface is a privately-financed Dutch company, with an established capability for the highly-productive delivery of enterprise software. Uniface Mobile:

- Satisfies a pressing business need for the addition of a mobile channel to existing Enterprise systems;
- Is a highly-productive, model-driven, component-based development environment, making good use of JavaScript and HTML5 in addition to its own scripting language;
- Is able to re-use and re-purpose existing business logic and data structures efficiently;
- Can leverage Uniface's strong and evolving partner community;
- and, should help to recruit new customers for the Uniface platform.

Key findings

In the opinion of Bloor Research the following represent the key facts of which prospective users should be aware:

1. Uniface provides an effective and speedy approach to the addition of a mobile channel to an existing business application.
2. Uniface has always provided excellent scalability, interoperability, vendor neutrality and portability, and has an excellent reputation for supporting its customers as they take on technology upgrades.
3. Uniface makes a distinction between business logic and application logic, which makes it especially appropriate for the addition of a mobile phone/tablet presentation layer on top of an existing business application;
4. The newly independent Uniface now has the resources and enthusiasm needed to secure its future, as it modernises from Uniface 9 to Uniface 10 and beyond.

The bottom line

Uniface Mobile is the first fruit from a revitalised, newly-private Uniface. Uniface can provide the highly-productive environment and new technologies needed to make this successful; but probably more important than its technology is Uniface's wealth of experience and the new fire running through the company. You can see Uniface 9.7 as a precursor of Uniface 10 Mobile Edition, but with the familiar Uniface 9 UI. It will let existing Uniface 9 customers make use of many of the Mobile capabilities developed for Uniface 10; and there will be a migration path to Uniface 10 Mobile Edition, which will have a fully modernised development environment. Uniface 10, like Uniface 9.7 offers high-productivity, browser-neutral, mobile development and deployment, making use of existing business logic wherever possible, but its new development environment will make coding even more productive, especially for developers coming from other tools/environments, and will reduce barriers to Uniface adoption.

At Bloor Research, we think that the initial Uniface promise, of efficient and speedy application modernisation through addition of support for a Mobile channel, will go down well. Uniface 9.7 is even planned to deliver such advanced Mobile features as installable apps for the various AppStores. Even so, the full realisation of Uniface's Mobile vision (including, in particular, offline working) will come with Uniface 10, later in 2015. We do think that Uniface will need to be careful to maintain clear reasons for its customers to move to Uniface 10 when the Enterprise Edition becomes available, or it will have a growing set of customers sticking with Release 9.7. Nevertheless, there is a good reason for even satisfied Version 9.7 customers to move to Version 10, as its modernised UI promises increased productivity for Uniface programmers, and reduced barriers for the recruitment of non-Uniface programmers.

Uniface Mobile

Implementation of Uniface Mobile

Uniface is taking an evolutionary approach to its target of a first-class mobile development platform, offering:

- Easy development (from data-source to device) of cross-platform, installable (which means, loosely, that they can be obtained from the appropriate App Store and appear on the home screen of the device), data-oriented, business-focussed, mobile apps;
- Native look-and-feel by default, which means that a Uniface app for the iPad, for example, should be indistinguishable in features, user experience and performance from a native iOS iPAD app;
- Hardware integration, which means full access to camera, GPS and so on;
- Software integration, which means complete adherence to all supported mobile platform standards;
- Off-line support, so business operation can be supported when connectivity fails and transactions, databases etc. reconciled when connectivity becomes available again.

The first stages in this journey are being delivered with Uniface 9.7, planned for September 2015: basic cross-platform GUI features such as headers, footers, slide-in panels, master detail etc., that work as expected on the appropriate mobile device. You can expect simplified development (with normal Uniface high-productivity); enhanced support for HTML5, custom properties on Uniface bound elements (such as Field, Occ and Ent), JavaScript promises and improved navigation (with field prompts).

However, in line with its focus in customer involvement in the development of Uniface, Uniface is thinking of making pre-release versions of Uniface 9.7 available to customers who can't wait. As well as reducing the risk of existing customers with an urgent need for Mobile development looking at other development platforms, this could be a valuable source of feedback and result in a better release candidate.

Apart from HTML5, the most exciting thing in this initial Release 9.7 from a programmer point of view is probably support for JavaScript promises, which allow developers to handle asynchronous events in JavaScript, a single-threaded language, without blocking execution flow (a promise can be "pending", the initial state; "fulfilled", i.e. successfully executed; or "rejected", i.e. failed; and once "failed" or "rejected" can never change again) and in a standard way. The stated Use Case for this first stage is: "as a supplier I need to allow potential customers to interact with my website in a mobile-friendly manner."

The second stage, possibly delivered with a point release of Uniface 9.7 if it doesn't make the first release in September 2015, will bring installable apps, that can be deployed through the appropriate app-store or side-load function. This has to include full hardware integration (for example, with camera, phone and GPS); and software integration (for example, with the contacts database and barcode scanner). A typical Use Case would be: "as a supplier of a SaaS expense system I must allow my users enter their expense claims along with a photo of the receipt and a GPS location."

The third stage is support for off-line operation; and probably involves the most new work, so it will be delivered with Uniface 10 Mobile Edition (again, available in Q3 2015). An off-line model (with application definitions and language) will be needed, with a data model supporting off-line local storage seamlessly, and automatic synchronisation with a master database when connectivity returns. A typical Use Case would be "as a hospital doctor on my rounds I do not have access to a network; however I still need patient data and anything I add to the record needs to be stored in the central records system".

At Bloor Research, we think that its mobile development capabilities are probably key to the potential acceptance of the Uniface v10 Platform as a first-class universal enterprise development platform, as opposed to Uniface's current status as a niche player.

The Uniface architecture

The current Uniface 9 architecture, still used in Uniface 9.7, follows a conventional 3-tier client/server model, with strong separation between business logic, data structures and presentation logic. To a large extent, this has evolved rather than been built from scratch for Uniface 9, but Uniface has a very disciplined approach and it is entirely suitable for adding a Mobile Presentation Layer to Uniface applications. Uniface has a well developed Modernisation strategy (see the Bloor White Paper at ***www.bloorresearch.com/research/white-paper/2156/uniface-modernisation.html***) which could be used to support this.

Uniface 10, however, is the result of a considerable internal modernisation program at Uniface, starting (in the days of Uniface 9) with the adoption of an Agile development process. The new Uniface 10 user experience is more than cosmetic – more than the just new IDE with its new graphical user interface (UI) and improved branding. More details of Uniface 10 can be found in the Uniface 10 InDetail paper from Bloor.

Uniface developers work in a structured framework, with a strong segregation between business and presentation logic.

Uniface 10 has a tabbed Single Document Interface (SDI), with each tab made up of several views and is "non-modal" (so a window can be left open while work continues elsewhere, for example). Programmers will largely work with HTML, CSS and JavaScript in the View component, especially with Mobile development, to define the User Interface, whilst writing the business logic largely in Uniface's own scripting language. View will display information to the developers, when the Controller tells View that the Model has changed; and will allow developers to interact with Uniface. The View component will then invoke the Controller back in Uniface on the Server to update the Model, which makes any changes persistent.

With Uniface 10, you can still think of a Development Architecture and a Deployment Architecture. Separation of Development and Deployment allows Uniface developers to build applications easily, quickly and productively, focussing on the business rules and requirements, without worrying about the deployment architecture, which can be optimised for Enterprise scalability, performance and security. It works, and it suits the needs of Uniface's customers.

Uniface Mobile Development

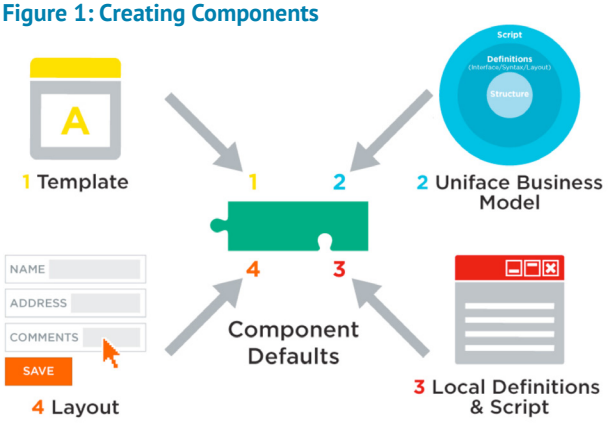
Bloor would agree with Uniface that the Uniface platform is extremely well suited to mobile development (in fact Version 9 already supported development of mobile applications well before Version 9.7). This is because of Uniface's high productivity, model-driven approach, its strict segregation between business-logic and presentation, and built-in enterprise-strength features (scalability and security, for example). The promise of the Uniface Version 10 platform is that with a new and more familiar (to non-Uniface developers) User Experience, you can use it to modernise an existing, not necessarily Uniface, business application, quickly and effectively (because Uniface can reuse existing data models and application logic), with a first-class (native look and feel) mobile device interface.

Uniface uses a team-based, Agile, model-driven, three-tier, component-based development (CBD) approach: modelling, construction and assembly. The process normally starts with modelling and Uniface's Graphical Model Viewer is well integrated with the environment and will probably be the user's first choice for building the model (although you can also import model definitions from third-party UML or traditional CASE tools, largely for legacy support).

Components are built starting with component defaults, modified with templates and then layout is added (**see Figure 1**). Next, the Uniface runtime deploys components and library objects, with supporting interfaces (**see Figure 2**); integrated under the Uniface runtime (**see Figure 3**).

The whole scales for the enterprise (**see Figure 4**), using the URB to link the Web components to the business logic and database/network connectors. The

Figure 1: Creating Components



Components are built starting with component defaults; which are then modified by combining the business model with templates for web pages, reports and services; and finally by adding the layout.

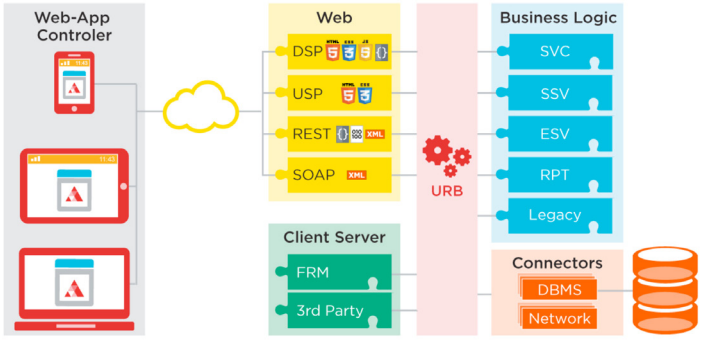
Figure 2: Deploying Components

The Uniface runtime deploys components and library objects and supports both HTTP and GUI interfaces and network and database connectors.

Figure 3: Assembling the Application

The Uniface application is then a tight integration of components under the Uniface runtime.

Figure 4: Scale for the Enterprise



URB – Uniface Request Broker
DSP – Dynamic Server Page
USP – Uniface Server Page
REST – Representational State Transfer
SOAP – originally Simple Object Access protocol
SVC – Uniface Service
SSV – Uniface Session Service
ESV – Uniface Entity Service
RPT – Uniface Report
FRM – Uniface Form
DBMS – Database Management System

dispatcher and broker can load balance incoming requests across the available resources.

Uniface is effectively object-oriented, with some departures from a purist OO model:

- It doesn't support a class hierarchy, although that doesn't limit what you can do in Uniface;
- In common with many competing tools, it doesn't support multiple inheritance; but many organisations find multiple inheritance very hard to implement effectively.
- It doesn't support dynamic binding, where properties are inherited at run-time, largely because Uniface believes that this impacts performance negatively and increases deployment and maintenance complexity. Uniface implements "object inheritance", where properties and changes to existing properties are inherited at compile time.

All Uniface deployment components (Forms, Services (SVC, SSV and ESV), Server Pages (Dynamic and static), and Reports) are compiled into tokenised objects, although they can be examined in interpreted mode for JAD (Joint Application Design – see ***http://en.wikipedia.org/wiki/Joint_application_design*** – prototyping) sessions. A powerful debugger is included.

Uniface Dynamic Server Pages (DSP) let the browser run asynchronous processes on the server (for database lookups etc.) and give developers a lot of flexibility in the way they handle the traffic between browser and server. Uniface manages potential 'race conditions' using declarative scoping; so that after the asynchronous DSP process is fired up, the user can be prevented from entering information locally which will be overwritten when the DSP returns.

Uniface has HTML editing capabilities, and due to the complete separation between the logic and presentation layers, it is possible to use an external HTML editor (such as Adobe Dreamweaver). It has a WYSIWYG layout editor, which is used to define layouts (including embedded data structures), by inserting text, images, links, tables, and so on; as well as for implementing more advanced features such as Java scripting and style sheets.

Dynamic Serve Pagers now communicate using JSON (Javascript Object Notation), which is easier for browsers to handle than XML and is claimed to be more human-readable (it is a lightweight standard using attribute–value pairs). JSON has become the usual standard for transmitting data between a server and web application. However, Uniface's XML and web services support still includes XML import and export, while Dynamic Server Pages can send an XML stream to a browser. There is also generic read/write functionality for XML data. You can simply import complex WSDL and automatically generate XML and/or database entities. Support for XML Schema (XSD) is included (this largely supercedes the older DTD mechanisms). Uniface also supports complex XML data types so that you can incorporate nesting, for example; and it supports disconnected record sets, which are needed for mobile and other similar environments.

Another potentially useful part of the product is the XSLT Workbench, which is a tool for creating and testing XSLT style sheets. Users can also perform XML transformations within the application, which is achieved by APIs, into XSLT functionality.

Support for SOA includes both provision and consumption capabilities, with the latter being achieved by the Uniface component employing a SOAP call-out connector that makes a SOAP request to the web service via HTTP. WS-I (Web Services Interoperability) compliance is supported. You can access SOAP headers from the Uniface script language; implement triggers (events) for XML call back; and use a struct data type, to

assist with sophisticated XML processing. The struct data type has now been extended to allow struct data to be passed between Uniface components using JSON as well as existing component data and XML. This lets developers move more complex data activities to dedicated services (possibly on dedicated servers) thus providing opportunities for performance optimisation and easier maintenance.

Advanced features of Uniface web applications include:

- Dynamic application-managed scoping and browser-side event handling, both reducing inefficient data exchange between client and server (resulting in less network traffic and server activity) and improving the user experience;
- Abstraction of transaction handling, allowing users to concentrate on the business logic;
- Bundling of open-source Tomcat 7, a more secure, better performing Java servlet engine than previously supplied (Tomcat is now installed with the Security Manager enabled, by default); this is good practice and ensures that it can be used in production Uniface web application deployments;
- Functionality for dynamically styling HTML elements thus (potentially) extending the user experience associated with Uniface applications.

Finally, Security is easily implemented in a Uniface environment, partly because the architecture itself is externally security-tested. However, it also has inbuilt protection against JavaScript. HTML and SQL injection; race conditions; and session management issues such as fixation and side-jacking. Uniface also offers inbuilt encryption routines, removing most of the risks associated with DIY encryption.

This Uniface environment is thus well-placed to deliver Uniface's vision for Mobile Development of cross-platform installable apps, with a single code-base supporting several mobile platforms. On previous form, Uniface should have no problem offering easy development from data-source to device; and its support for HTML5 will help it deliver native-like look-and-feel by default. It promises full hardware and software integration with mobile phone and tablet platforms and we anticipate no problems delivering an effective off-line experience in due course.

Legacy modernisation process

Legacy modernisation is particularly attractive as demands from businesses to "do more with less" proliferate and a separate white paper on a legacy modernisation methodology that can be used with Uniface (although it is applicable, in essence, in the general case) is available from Bloor Research (at ***www.bloorresearch.com/research/white-paper/2156/uniface-modernisation.html***).

Adding a Mobile channel to an existing application (either Uniface or non-Uniface) provides an easy win for modernisation with Uniface Mobile. You can look at adding rich localisation to application interfaces (appropriate country defaults and use of the local language) taking advantage of the fact that a mobile device generally knows where it is. Uniface has full Unicode support which allows characters from different languages (including Japanese, Chinese, Russian etc.) to be handled at the same time (that is, treated as just one huge character set). This is an advantage for the design of universal end-user data entry modules, even if only one language at a time will be displayed.

You can also add encryption, to protect information on an end-user device, as well as in transmission over the Internet. The Uniface documentation includes security guidelines for the development of applications for the web. Security is a subject that is continually evolving, and Uniface has set out to provide a framework for developers to work against, especially those migrating from client/server to web application development.

Uniface differentiators

Uniface has been around for longer than most of its competitors, servicing enterprise-scale applications, yet has managed to keep itself up-to-date as technology evolves. Moreover, Uniface has demonstrated its ability to take its customers along with it as its technology evolves, instead of abandoning them in an "old technology" ghetto. Uniface Mobile uses the latest Mobile technology approaches but hasn't abandoned any of Uniface's core values.

Another differentiator is its excellent back-end connectivity and capabilities for re-use of existing data structures – an Excel spreadsheet can be used as the source for a Uniface business application (which will, of course, be far more secure and more robust – and more maintainable – than the original spreadsheet).

Uniface Licensing models

Uniface licensing is managed with the Uniface Distributed License Manager (DLM). This is a server which can supply licenses to clients on request, so they don't need to be held locally. This is popular with Uniface's VAR customers and Deniz Yugnuk (Head of Sales, which includes Licensing) has even had some requests that it be productised, so that VARs could use it to manage their own user licensing, and the availability of particular features within their applications. Uniface hasn't decided whether to do this yet.

Uniface runs a range of very flexible licensing models but can't offer "pay per usage" models yet (it would like to, and is looking at these; but they are on the road map rather than in development).

Uniface Application Servers are licensed per server, per CPU per core, and this also applies to Web Application Servers. If (as with Uniface 10 Web Edition) you are developing only Web or virtualised Cloud applications, there are no runtime charges, you just pay for the Web Application Servers you need (and also for the development licenses needed to build the application before deploying it). Trial licenses can be downloaded from the Web.

Bloor's opinion is that Uniface licensing is flexible enough in practice but that "pay per usage" licensing with its associated agreed and transparent metrics, is the way forward, especially in web and virtualised environments.

VAR's and business partners just pay negotiated royalties to Uniface and can license their products to their customers in any way they like.

Supporting products – for partners and education

> **The Uniface web presence is supposed to be fun and welcoming to the new partners and developers Uniface will need in order to survive and prosper.**

Uniface has long had supportive on-line educational resources. In addition to these it runs several educational programs, intended to address any perceived issues with finding Uniface skills:

- The Uniface University Program, initially being rolled-out in Europe and US to selected Universities;
- In-house or Uniface-based training & mentoring programs;
- Uniface Workshops;
- The Uniface Academy.
- Uniface2Web is a new initiative, where web technology overview evening courses given to several customers have been recorded and will be made available to a wider web audience.

This is as would be expected for any responsible enterprise vendor but after experimenting with online portals, under the overall leadership of Aad van Schetsen (its President & General Manager), Uniface is now implementing a rich public-facing directory at **www.uniface.com**. This will be open to all Uniface stakeholders, which is generally a good thing, although we think it is a pity that there isn't also a private forum where Uniface stakeholders can discuss commercially sensitive issues under "Chatham House rules".

More than this, however, the Uniface web presence is supposed to be fun and welcoming to the new partners and developers Uniface will need in order to survive and prosper.

Uniface Mobile, the customer stories

The company has always kept faith with its customers and partners, allowing them to move from monolithic applications and character-based interfaces into Client/Server applications, GUIs and now to web and mobile applications. Many vendors simply abandoned its user-base over the Client/Server revolution; Uniface didn't. This approach survived the acquisition by Compuware in 1994 and, now, the sale to Marlin Equity Partners in 2014 – and will continue on into the future. Given the pressures involved in making modernisation decisions and the danger of making the wrong strategic choices if they are forced or hurried, this is an important part of Uniface's story as its customers now move into Mobile applications development.

Use Case 1: a Mobile-first website

The immediate Use Case for Uniface mobile is simple modernisation of an existing application or website with a mobile channel. This will satisfy requests such as: "as a supplier I need to allow potential customers to interact with my web-site in a mobile-friendly manner."

In other words, this Use Case describes customers or employees who want to interact with an organisation's website using either an Android or iOS smartphone or tablet (at least), with similar functionality to their existing Windows client or browser access and a user experience that is similar to that they are used to with other Smartphone apps.

Uniface can deliver immediate benefit here because it allows mobile apps to reuse existing enterprise assets (both business logic and data), so that any new mobile application assets can be strongly integrated with existing Enterprise back-end systems at little extra cost. This gives Enterprises an immediate Mobile channel with a familiar look-and-feel, which (because of Uniface's support for HTML5 and JavaScript etc.) can be browser and platform neutral – an immediate win, even though the mobile channel will not have significant functionality beyond that in alternative channels.

Use Case 2: Installable apps

As customers or employees become comfortable with using Smartphones or tablets for routine business, their expectations will rise. Users will want to use the more advanced capabilities of the

smartphone (such as GPS-based location services, the camera, and so on), and download their Apps from an appropriate AppStore. Uniface will have to deal with requests such as: "as a supplier of a SaaS expense system I must allow my users to enter their expense claims along with a photo of the receipt and a GPS location."

Satisfying this use case will exploit Uniface's ability to reuse existing application assets as in Use Case 1, of course, but then Uniface will provide, in time, access to all the features of smartphones, with a native "look and feel" and performance levels.

At this stage, mobile apps will provide features beyond those available to browser or desktop-based interfaces – how many portable PCs have a built-in GPS? This Use Case is likely to represent the normal way of accessing many business applications, in a year or so.

Use Case 3: Off-Line Apps

The big issue with mobile applications in business, for the foreseeable future, will be the reliability, resilience and availability of mobile coverage. WiFi is not yet available everywhere and high bandwidth phone coverage can't be guaranteed. Waiting to update business systems until you next have coverage is unacceptable for many business applications, so mobile applications will need to be able to operate offline and re-synch to the "Master" systems in the enterprise, including possible conflict resolution/negotiation, when a network or phone connection is next available. This is far from trivial as (in particular) a necessary subset of the Enterprise databases will need to be cached locally, for when connectivity is unavailable.

A typical Use Case here might be: "as a hospital doctor on my rounds I do not have access to a network; however I still need patient data and anything I add to the record needs to be stored in the central records system". Satisfying this Use Case will be a later delivery in the Uniface Mobile roadmap.

Many Uniface customers will be new to developing for Mobile devices and will need to bring their own development cultures up-to-date with Mobile. Access, through Uniface's web presence, to other customers that have made the change, could be part of achieving this.

Uniface BV, the company

Uniface has a strong reputation for supporting its customers and bringing them up-to-date with the latest technologies.

Uniface vendor background

Uniface B.V. has a long history, over several decades, since its initial inception by Inside Automation in the Netherlands, as a high productivity 4GL, It became Uniface B.V., back in the 20th century. It was then acquired by Michigan-based Compuware Corp in 1994, but product development continued in Amsterdam and it has recently been sold once again, in 2014, to Marlin Equity Partners. It is now Uniface B.V. again, a privately-funded Dutch company. A full description of the new Uniface BV can be found on the Bloor website at ***www.bloorresearch.com/supplement/uniface/***.

Customers

Uniface currently (2014) has about 1600 customers; and 250 partners and resellers – who are also treated as customers. Uniface is currently building a strong global partner organisation, with local partners that want new customers themselves – and which have local knowledge and networks in places like China and South America. There is a long list of Uniface customer case studies at ***www.uniface.com/customers/#1***.

Customers trade with Uniface because, firstly, it is far more productive of business outcomes than conventional development environments such as Java (Eclipse) or .NET (Visual Studio). Secondly, Uniface has a strong reputation for supporting its customers and bringing them up-to-date with the latest technologies, such as Mobile, at their own pace.

Competitors

Uniface will face strong competition in the mobile space from new high productivity PaaS vendors who have grown up with Mobile. However, such competitors may not always have its established longevity, nor its demonstrated capability for the support for, and integration with, enterprise back-end systems.

Partners

Uniface Partners are an essential part of Uniface's new go-to-market strategy, as exemplified in its Partners United Program. It has seen a 13.8% growth in Application Partners business in its 2014 financial year and it has, it says, improved its working relationships with its key partners and is making new commercial opportunities for its VAR partners.

Global partners include VBT (Turkey); Wizrom (Romania); Everest Computers (UAE & Dubai); Mobilne (Croatia); Techshire and TCS (India); ONE1 (Israel); Hongyi (China), Mainsoft (Brazil, Peru, Chile); Softline (Russia). They obviously give Uniface a global reach and access to local business networks, without Uniface itself having to build a presence in each country. This approach also helps to insulate Uniface from any political issues with, for example, doing business with an EU company in Russia. Since Mobile is a primary rout to banking in some areas of the world, this focus on global partners with their own local customer networks and relationships should open opportunities for its new Mobile capabilities.

Uniface is also, now it is independent of Compuware, actively rebuilding its Channel.

Financial information

Uniface is owned by Marlin Equity Partners (MEP), which is a global private equity firm operating throughout North America and Europe. Since 2005, it has completed over 70 acquisitions and currently has over $2.6 billion of committed capital from its blue chip institutional investor base. Its other successful growth acquisitions have achieved from 25% to over 400% growth under MEP.

Current issues

In the context of Mobile development, Uniface has to make a name for itself and mustn't let itself fall behind the competition. Its staged approach seems likely to help it address this and provide immediate support for its existing customers; but it will need to be careful that the availability of Uniface 9.7 doesn't "steal Version 10's thunder" and unduly slow the migration of customers to Version 10. Uniface, however, seems well aware of this potential issue.

Summary

Uniface has seen the opportunities available for Mobile development and moved reasonably fast to take advantage of them. Its staged approach to making a mobile development capability available to existing Version 9 customers is a good one, as long as it makes Version 10 a sufficiently compelling upgrade for mobile developers for them to move forward in a reasonable time-frame.

Uniface's key offering to mobile developers is its strong (through model-driven development) separation of the presentation layer, business logic layer and data layer; and its established back-end connectivity features. This means that it is extremely well-placed to add a mobile channel to existing enterprise applications (even those not currently written in Uniface) whilst efficiently re-using existing application and data assets.

Its attention to customer welfare and enterprise scalability has been demonstrated over a very long period, making it probably a lower-risk approach to building a mobile channel than some younger products.

FURTHER INFORMATION
Further information about this subject is available from
www.BloorResearch.com/update/2243

About the author

DAVID NORFOLK
Practice Leader / Development and Governance

David Norfolk first became interested in computers and programming quality in the 1970s, working in the Research School of Chemistry at the Australian National University. Here he discovered that computers could deliver misleading answers, even when programmed by very clever people, and was taught to program in FORTRAN. His ongoing interest in all things related to development has culminated in his joining Bloor in 2007 and taking on the development brief.

Development here refers especially to automated systems development. This covers technology including acronym-driven tools such as: Application Lifecycle Management (ALM), Integrated Development Environments (IDE), Model Driven Architecture (MDA), automated data analysis tools and metadata repositories, requirements modelling tools and so on. It also covers the processes behind them and the people issues associated with implementing them. Of particular interest is organisational maturity as a prerequisite for implementing effective (measured) process and ITIL (v3) as a framework for automated service delivery.

David is a past co-editor (and co-owner) of ***Application Development Advisor*** and associate editor for the launch of ***Register Developer***, and is currently executive editor for GEE's "IT Policies and Procedures" product. He has an honours degree in Chemistry and is a Chartered IT Professional, has a somewhat rusty NetWare 5 CNE certification and is a full Member of the British Computer Society (where he is on the committee of the Configuration Management Specialist Group).

His early career involved working in database administration (DBA) and operations research for the Australian Public Service in Canberra. David then returned to his UK birthplace (1982) where he worked for Bank of America and Swiss Bank Corporation, at various times holding positions in DBA, systems development method and standards, internal control, network management, technology risk and even PC support. He was instrumental in introducing a formal systems development process for the Bank of America Global Banking product in Croydon.

In 1992 he started a new career as a professional writer and analyst. Since then he has written for many major computer magazines and various specialist titles around the world. He helped plan, document and photograph the CMMI Made Practical conference at the IoD, London in 2005 and has written many industry white papers and research reports including: IT Governance (for Thorogood), Online Banking (for FT Business Reports), Developing a Network Computing Strategy and Corporate Desktop Services (for Business Intelligence), the Business Implications of Adopting Object Technology (for Elan Publishing).

He has his own company, ***David Rhys Enterprises Ltd***, which he runs from his home in Chippenham, where his spare moments (if any) are spent on photography, sailing and listening to music.

Bloor overview

Bloor Research is one of Europe's leading IT research, analysis and consultancy organisations, and in 2014 celebrated its 25th anniversary. We explain how to bring greater Agility to corporate IT systems through the effective governance, management and leverage of Information. We have built a reputation for 'telling the right story' with independent, intelligent, well-articulated communications content and publications on all aspects of the ICT industry. We believe the objective of telling the right story is to:

- Describe the technology in context to its business value and the other systems and processes it interacts with.
- Understand how new and innovative technologies fit in with existing ICT investments.
- Look at the whole market and explain all the solutions available and how they can be more effectively evaluated.
- Filter 'noise' and make it easier to find the additional information or news that supports both investment and implementation.
- Ensure all our content is available through the most appropriate channel.

Founded in 1989, we have spent 25 years distributing research and analysis to IT user and vendor organisations throughout the world via online subscriptions, tailored research services, events and consultancy projects. We are committed to turning our knowledge into business value for you.

Copyright and disclaimer

2nd Floor
145–157 St John Street
LONDON EC1V 4PY
United Kingdom

Tel: **+44 (0)20 7043 9750**
Web: **www.Bloor.eu**
email: **info@Bloor.eu**