

User experience: Web widgets vs. Web browsers

Mobile phones are characterized by their software and hardware limitations, like display size, processing power, interactivity and connectivity. As a result, deployment of full-blown browser-based Web applications on the handset may not be the most efficient or the most cost-effective way to offer an enhanced Web experience; fig. 3.15 lists the main challenges facing Web browsers and the advantages of deploying widgets as means of improving the mobile Web experience.

Figure 3.15: Problems with full mobile Web browsing and the widgets value proposition

Full Web browsing	Widgets value proposition
Need to launch a Web browser to access online resources.	Lightweight applications that can continually update specific information and display it on handset's screen.
Interaction with Web application could be challenging in a mobile context where the user is moving.	Multiple widgets can run simultaneously on the handsets, provided the handset supports multitasking.
Inefficient approach to discover and access regularly-updated information and Web services like e-mail, news papers, information, social networking and infotainment services.	By combining Web data with context information of the user and the handset, e.g. using proximity detection and location, it is possible to deliver context-based information to the user, without requiring extensive input from end users.
User experience is seriously affected by disconnections from the mobile network.	Allow continuous access to the service despite intermittent disconnections.
Higher charges since the end user needs to search information source first then locate it and load the full Web page.	Represent an ideal mean to convey accurate Web information in bite-size format
	Appeal to cost-conscious customers.

Source: Informa Telecoms & Media

Widgets have also advantages for WAP services and native mobile applications (see fig. 3.16). It is clear that widgets can successfully substitute WAP services and could provide a better user experience, richer content and more personalized and relevant services. Widgets can also replace simple native applications where time to market is a priority.

Figure 3.16: Comparison between mobile widgets, WAP and native mobile applications

Features	Widgets	WAP	Native applications
Customization	(+) Easy customisation of the UI which can be done after deployment	(-) Limited	(-) Customisation needs to be carried out before deployment
Network dependency	(o) Depends on network, but can work offline too	(-) Totally dependent	(+) Not dependent
Idle screen integration	(+) Yes	(+) Yes	(+) Yes
Device APIs support	(o) Depends on widget runtime, and OS vendor/OEM making APIs accessible	(-) No	(o) Depends on execution environment, and OS vendor/OEM making APIs accessible
Network APIs support	(o) Depends on widget runtime and operator making APIs available	(-) No	(o) Depends on execution environment and operator making APIs available

Notes: (+) = positive; (-) = negative; (o) = neutral. * most current WRTs run over feature-rich phones but very rapidly encroaching on to the low-feature phone segment.

Source: Informa Telecoms & Media

Mobile Web runtime market and competition

Operators and handset vendors are in a race to create or adopt the best MWRT platforms that fit best into their mobile Internet strategy. A number of MWRT solutions have been proposed over the last two years by:

- Global software vendors, e.g. Adobe AIR and Microsoft Silverlight.
- Global Internet service providers, e.g. Yahoo! Blueprint and Google Gears.
- Handset vendors, e.g. Nokia WRT, Motorola WebUI, Samsung Mobile Widgets.
- Mobile browser vendors, e.g. Opera Widgets and Access Netfront Widgets.
- Mobile software solution providers, e.g. SurfKitchen SurfKit, Ikivo Enrich and Novarra Vision.

Figure 4.2: Mobile Web runtime providers and their customers

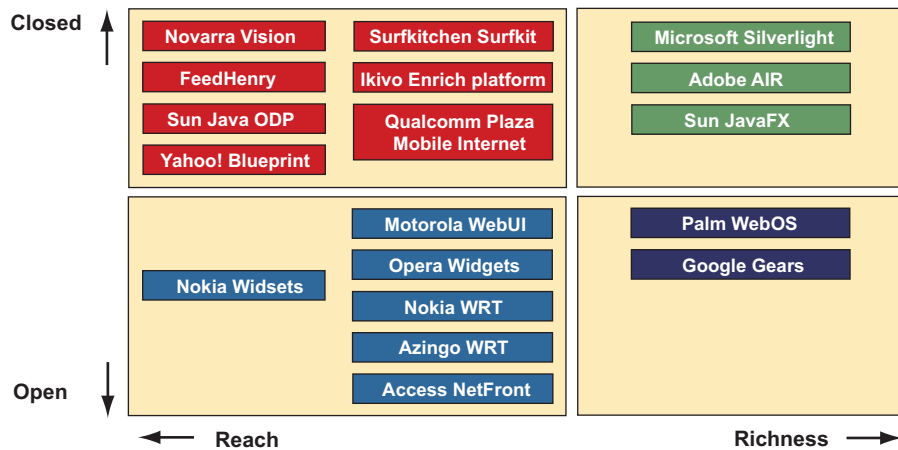
Company	Products	Widgets	ODP (inc. app storefront)				Clients
				AIS	WS-WL	WS-CR	
Adobe	AIR Mobile*	✓			✓		Not commercialized yet
Access	Netfront Widgets WRT, Netfront Dynamic Menu	✓	✓	✓	✓	✓	NTT DoCoMo, Willcom, Softbank, eMOBILE, Disney Mobile.
Azingo	Azingo WRT, Active HomeScreen	✓		✓	✓		Undisclosed
FeedHenry	FeedHenry	✓	✓				Munster Express and Eircom
Google	Gears	✓				✓	Google, Zoho and Buxfer
Ikivo	Enrich	✓	✓		✓		Vodafone, GNT (Japan)
Microsoft	Internet Explorer Mobile 6 Widgets	✓					Preloaded with Windows Mobile 6.5+. Not commercialized yet
Microsoft	Silverlight Mobile*	✓			✓		Not commercialized yet
Motorola	WebUI	✓		✓		✓	Deployed on Motorola MOT VE66, ROKR EM35 and EVOKE by Telus Wireless, Alltel, Cricket Communication, Cellular South
Nokia	S60 WRT	✓		✓		✓	Deployed on some Nokia, Samsung and LG handsets
Novarra	Vision Platform	✓	✓				Undisclosed
Opera	Widgets	✓		✓		✓	KDDI, Vodafone, T-Mobile
Qualcomm	Plaza Internet Mobile	✓			✓		America Movil
Samsung	Touch Wiz UI + Samsung Mobile Widgets	✓		✓		✓	Global distribution
SurfKitchen	SurfKit	✓	✓	✓	✓		Telstra
Yahoo	Yahoo Blueprint – Yahoo Go	✓			✓		Freely downloadable from Yahoo Website**

*Notes: *This widget runtime primarily supports proprietary technologies but can also render Web technologies. **Yahoo Blueprint application can be deployed as: a Web site accessible with a Web browser; a standalone application for Java, Symbian or Windows Mobile; or as a widget for Yahoo Go 3.0 client which is supported on 330 phone models from various OEM*

Source: Informa Telecoms & Media

various widget types from those supporting basic functionality to those packed with innovative features.

Figure 5.1: Mobile widget runtime ecosystem quadrants and vendor positioning



Source: Informa Telecoms & Media

Vendors that offer proprietary MWRT solutions will need to continue to innovate in order to remain competitive with those that have opted for an open approach. Informa Telecoms & Media believes that openness will be a key requirement of operators and an influential factor in their choice of an MWRT solution. Open platforms are likely to enable the development of a variety of widgets targeting various segments of the mobile handset market. Openness through the use of Web standards will also lower the entry barrier to application developers and enable the creation of a plethora of low-cost applications. Tools for developing Web applications are much easier to use, which means that, by using Web-based WRTs, developers can lower the cost associated with applications development and shorten time to market for application distribution and monetization. In addition, the extended reach of these platforms to target the wider Web developer community could enable the creation of a long tail of applications targeting various consumer groups.

However, although limited in scope and reach compared with open solutions, closed MWRT platforms will continue to be deployed as they provide unique advantages to operators and OEMs, including better control over the product development roadmap, flexibility for customization and integration as they are designed specifically for the infrastructure and devices for which they are intended.

For these reasons, Informa believes that there will be market opportunities for both open and closed approaches and operators' choice will depend on the market segment they want to address and the business model they will deploy for offering widget services.

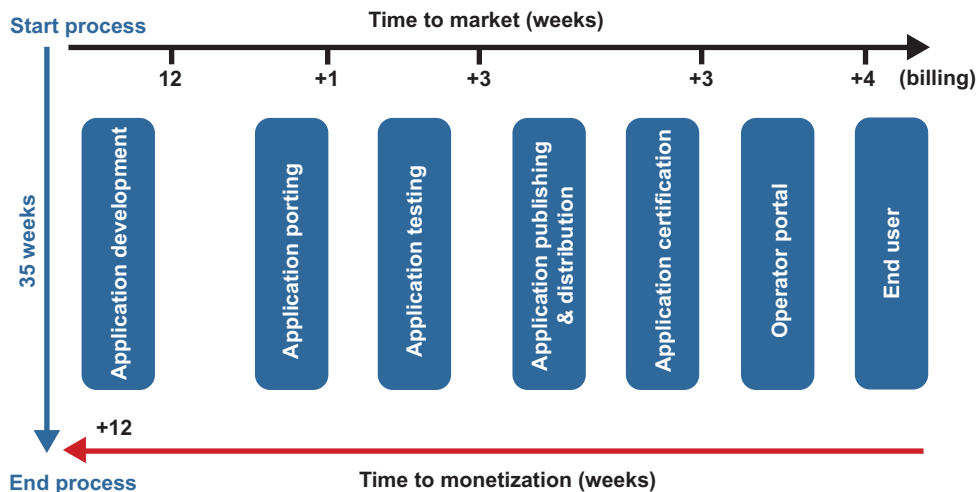
be compiled into machine code(i.e. translated into the language to be understood by the device), this step is not specifically required for Web applications as the languages used are directly interpreted to the machine language.

Relationship of operators with developer community

Unlike the desktop market where the open ecosystem has largely matured, the mobile environment has traditionally been a closed ecosystem largely controlled by mobile operators. It has not been easy for mobile application developers to grow and establish themselves in such an environment. In the mobile value chain that feeds content to operator portals, developers have been always right at the bottom, and the revenue from the sale of the content they create tends to filter down through several middle layers before the last – and often the smallest – portion ends up in their hands. The biggest share is kept by the operator, which sells the content to end users and then there is often more than one intermediary, such as an aggregator and a publisher (depending on the type of content), that takes its share. What is left for the developer can often be as little as 10% or less despite the fact that the developer is often the one that foots the bill for the creation of the content, and in mobile such costs can be huge – especially for graphical, fully multimedia applications, such as games.

Making content for operators means the developers have to create numerous versions to cater for as many as they can of the hundreds of handset models that operators distribute – with their different operating systems, flavors of Java, browsers, media players, and screen sizes and resolutions. Mobile application developers are stymied by the extreme difficulty of writing mobile apps for multiple device operating systems and runtime environments, especially when there is no clear winner and diversity is increasing. Porting the mobile application to reach a larger addressable market can break the business model of the individual developer – if there is one – making applications development a very difficult market to be in.

Figure 6.2: Application development cycle and time to monetization from the mobile operator portal



Source: Informa Telecoms & Media

Figure 8.3: Evaluation of mobile widget ecosystem requirements as a function of OEMs' business models

Objectives	Mobile widget ecosystem requirements	Business model
1, 2 and 3	Support for Web services	Brand differentiation
	Support for Web technologies	Customer retention
	Enable active idle screen integration	Reducing churn
4	Support for device APIs	Value-added services: context-based widgets and applications
5 and 6	Support for network APIs	Double-sided business models (e.g. selling APIs to third parties, ad- and commercials-based apps)
	Support for ODP (inc. application storefront)	Service enablement
7	Support for third-party developers	Rich Internet application
		Long-tail business model
		Spur innovation
8 and 9	Integration with application store	Content retail (application stores, Web stores)
	Support for billing	
10	Deployment of hybrid solutions (native and Web-based)	Pricing models and consumer segmentation
		Service plans and service brokering

Source: Informa Telecoms & Media

A single mobile widget platform may not be able to satisfy all of the requirements set by the operator and so could be restricted to support only particular business models. In this case, an operator could combine different products incrementally that are in line with various business models. For example, the Web user experience is key differentiating point for operators aiming to improve their brand image associated with Web services, and it is a core capability of all mobile widget runtimes. Using widget solutions to enable intuitive and easy-to-discover Web services significantly enhances the overall user experience if these solutions provide the user with the flexibility to customize and personalize widgets in the idle screen. The customers can then relate their experience to the operator brand which could significantly increase customer loyalty and help the operator to reduce churn and attract new customers.

Most Web-based mobile widget runtimes provide access to device APIs, given that the underlying OS exposes these APIs to third-party developers. This way, widgets can make use of devices features and services, including the address book, GPS, photo and music albums, camera, calendar, and PIM, to provide a richer experience and context-based functionalities. Not only does this have the potential to enrich user experience by presenting local content and handset functionalities in a totally new way but could also enable the creation of new value-added services for enhanced ARPU.

This experience could be further enhanced if the runtime can take advantage of the multitude of the network assets through the use of network APIs. This approach has the advantage of combining operators' services, such as sending an MMS or SMS, video telephony or location, with widget capabilities to create a new type of combined services offered either by the operator or by a third-party service provider. When a third-party service provider is allowed access to the operator's subscribers, revenues are generated using the so-called two-sided business model, under which the subscriber is charged for the actual traffic generated and