



Technical white paper

A technical whitepaper on Mobility-as-a-Service - MobiDM

A mobile device is a part of the modern mankind and thus no mobile device is the same. Organizations of all sizes are increasingly challenged to secure and manage employees' mobile devices, including applications and data residing on them, without having to build, install and maintain their own solutions.

Delivering Mobility as a Service eliminates the need for knowledge, hard- and software and does not require investments upfront.

January 2010

Introduction

The 'cloud': a term that has featured frequently in numerous articles over the past year. It is used as a metaphor for internet-based development and usage of computer technology. Mobile devices are wirelessly connected within this cloud also, and as they travel along with their owners it makes them hard to manage and secure.

Device-management-software can help, but requires upfront investments in expensive servers, software, administrators, databases and all kind of licenses. On top of that it usually only covers one kind of brand or type of mobile device. In today's world it is the mobile end-user and not the IT department that decides what devices will be used, making it even harder to manage and secure the phone-platform jungle.

Using SaaS helps to keep all of the above components behind the scenes, allowing the user to focus on other things: his or her actual job. Applying a SaaS model and extending it to cover most operating-systems available, and thus providing a robust platform to manage and secure the mobile devices, requires the use of the best of breed software, such as Sybase's Afaria, MS Active Directory, MS SQL-Server and proven technologies like virtualization, XML, master/slave, load-balancing, AJAX and SOAP.

Delivering the above as a platform has given rise to a new service: Platform as a Service (PaaS). Focusing this service on the mobile workforce creates a new form: Mobility as a Service.



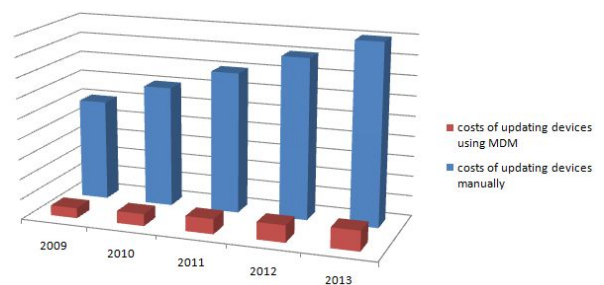
Device management and security

Mobile device management (MDM) refers to any routine intended to distribute applications, data and configuration settings. The intent of MDM is to optimize functionality (manage the device) and security of a mobile communications network, while minimizing cost and downtime, in a way that is as transparent to the end-user as possible.

Device Management

Device Management (DM) covers the tasks to be executed on a mobile device, such as installing applications and altering system settings, typically performed by the IT department's administrator.

As this figure shows, the costs of managing these devices will increase significantly, if MDM is not used. As the administrator needs to stay in control, it is important that the end-user is not capable of overwriting these actions. DM should be treated in the same way as the traditional



management of desktops and laptops.

Security

Enforcing pin codes or passwords and encryption of files, documents and e-mail is an important aspect, as it creates a unique key-combination between the device and the files or PIM-data stored on it, either in the built-in memory or any external SD-card or USB stick. Encryption should be done on the fly. Mobile Device Security can be applied on different levels. Always keep in mind that it might affect the usability of for the end-user. If users need to enter their pin code to unlock the company's phone every two minutes, the chances are that they will use another unsecure (personal) device with all consequences of this. We strongly advise preparing a fixed set of predefined security levels that have been carefully set up, which can be chosen from.

Multi-OS

In today's world it is the mobile end-user and not the IT department that decides what devices will be used. This results in the need to manage all kind of different mobile device brands and types, with their respective operating systems. Changing a simple thing like the device's backlight settings can be completely different between an iPhone and a Nokia and this all needs to be clear to the administrator. After all, more and more people just use a phone, and do not care – or even know – what OS the phone is using. And they shouldn't need to know: the system used for managing the devices should do this.

Platform Market Share — 2012				
2012 Preliminary Forecast – 522 Million Units				
Platform	Unit Sales (M)	4Q/12 Share	1Q/09 Share	Difference
Symbian	203.58	39.0%	49.3%	-10.3%
Android	75.69	14.5%	1.6%	+12.9%
iPhone OS	71.51	13.7%	10.8%	+2.9%
Windows Mobile	66.82	12.8%	10.3%	+2.5%
RIM OS	65.25	12.5%	19.9%	-7.4%
Linux	28.19	5.4%	7.0%	-1.6%
webOS	10.96	2.1%	0%	+2.1%

As of October 2009 Gartner

Mobility as a Service

To offer a wide spectrum of mobile features as an online service, the solution built should go beyond the characteristics of SaaS or PaaS (Platform as a Service). SaaS is a model of software usage where the vendor provides a solution that customers use as a service-on-demand, typically by using the internet. Customers pay a subscription or a usage-related fee for using the application. SaaS vendors are responsible for hosting and maintaining the application, so clients only need Internet access and (where appropriate) a web browser to use it, reducing the overall IT costs to the customer. PaaS is a computing solution-stack-as-a-service. It goes further with the provision of a software development platform, which is designed for cloud computing at the top of the cloud stack.

Software as a Service – SaaS

Ready to use applications that can be rented

Platform as a Service – PaaS

Development platform which includes a virtual infrastructure and OS, on which can be developed

Infrastructure as a Service – IaaS

A platform for deployment with a virtual infrastructure

PaaS facilitates the deployment of applications without the cost and complexity of buying and managing the underlying hardware and software, providing all the facilities required to support the complete lifecycle of building and delivering web applications and services entirely available from the Internet. This is generally known as cloudware. PaaS options typically include workflow facilities for application design, application development, deployment and hosting as well as application services such as web service integration, database integration, security, scalability, storage and persistence.

Mobility as a Service (MaaS) focuses on all aspects of a online service, where it goes beyond a PaaS, towards being a IaaS (Infrastructure as a Service). It is able to manage the entire infrastructure or life-cycle of the mobile workforce, from the out-of-the-box deployment until its last kill-pill.

Best practices

To ensure that the Mobility as a Service will answer the needs from the customers and survive in the platform jungle, some best practices should be kept in mind.

- Use one front-end for the website's functionality.
To make the MDM's control panel accessible to both the admin and the end-user a web portal is ideal as it is available everywhere you have internet access, even on the mobile devices itself. The only prerequisite is having a browser.
- Support multiple device types and OS's.
Workers outside in the field, often in rugged environments, are well suited for Windows Mobile-powered smartphones, whereas information workers often demand BlackBerry, Nokia E-series powered by Symbian, or iPhone devices for personal information management. It's important for IT professionals to know their mobile workforce to ensure their users receiving the form factors, devices, applications, and levels of security that enable them to be most. Critical is to invest in a mobile device management solution that supports multiple platforms enterprise wide, so you're not locked into a single platform for your diverse user needs across multiple departments or lines of business.
- Support a mix of business and personal use.
It has quickly become a necessity for firms to allow the dual usage of mobile devices that contain both professional and personal profiles, data, and applications on a single device. Application-level or file-level security (including a strong password policy and encryption) will allow personal usage to continue without change, while allowing IT to securely manage corporate data.
- Encourage a strong password policy.
Because of mobile devices' inherent vulnerability to misplacement and theft and their always-on status; it's critical that firms protect themselves through a strong

password policy. All devices should be protected through strong passwords that require a refresh after between 3 and 6 months in use. Additionally the companies with which we spoke also enforce remote lock after a specific period of non-use. Interestingly, the time period ranged from as little as 3 minutes to as long as 72 hours

- **Encryption.**
Consider file-level, application-level, or full-disk encryption to secure sensitive data and provide legal safe harbour from disclosure requirements in the event of lost or stolen mobile devices or removable media cards. Also ensure that all sensitive and confidential data is encrypted when it's in transit over the air between mobile devices.
- **Enable remote or automated lock or wipe.**
Because devices are often left behind in taxis or stolen, it's important that IT has the capability to send a remote lock or wipe command to protect the data that resides on the mobile device. In the event that a user simply misplaces their device temporarily, users feel reassured knowing that their person data will remain protected until they find their device again and unlock it or that they can fall back on wiping it if they fail to retrieve it.
- **Backup and Restore**
No management platform is complete without the capabilities for automatic backup and recovery of settings, files, and applications. This is particularly important given how often devices are dropped onto concrete sidewalks or replaced due to theft or loss. Moreover, backup and recovery can help with rapid transition to new devices, a common occurrence in today's volatile device market that often sees quick refresh cycles of one to two years.

Open standards and components

An online service-platform can be a complex system of different objects that all need to communicate with each other. Focus on using proven state-of-the-art technology is crucial to ensure a stable system for all customers.

AJAX

A web based portal should be used to manage the devices and should support the most commonly used browsers like Firefox and Internet Explorer. AJAX technology can significantly increase the user experience, as it makes the website look and feel like a desktop application. It's shorthand for Asynchronous JavaScript and XML and is a group of interrelated web development techniques used on the client-side to create interactive web applications. With AJAX, web applications can retrieve data from the server asynchronously in the background without interfering with the display and behaviour of the existing page. Data is usually retrieved using the XMLHttpRequest object.

SOAP

A common method used in communications between different systems is SOAP (Simple Object Access Protocol). It's a protocol specification for exchanging structured information in the implementation of Web Services in computer networks. It relies on Extensible Markup Language (XML) as its message format and makes use of an internet application layer protocol as a transport protocol. Some advantages, especially for PaaS solutions:

- It is an open standard
- Using it over HTTP allows for easier communication through proxies and firewalls
- it is platform and language independent

OMA CP/DM

OMA DM uses XML for data exchange and allows device management by communication between the server and the client. Most mobile devices already have the OMA-client-software already built-in when leaving the factory, making them manageable out-of-the-box.

OMA DM is designed to support and utilize any number of data transports such as GSM, Bluetooth, WAP or HTTP. The communication protocol is a request-response protocol. The OMA DM server, using any of the methods available such as a WAP Push or SMS, initiates the communication.

Although OMA is open, it's not fully supported on all devices. Different brand have implemented their own versions (free-form) and only accept specific type of OMA messages.

Multi-tenancy

Multi-tenancy refers to a principle in architecture where a single instance of the software runs on a server, serving multiple client organizations (tenants). All features in the portal are (depending on the license) available for all customers, and for efficiency purposes they are – in contrast to data storage- not physically copied to the customer's own part of the portal. They there for use common resources and functions which should be kept apart strictly and secure.

The principle of multi-tenancy is not universally accepted and supported within the software industry, but is for sure one of the key-factors where one can excel in.

Extensions

As the common MDM- and Security are becoming commodity functions, a MaaS needs to extend into more directions in order to increase functionality. These 'Extensions' add extra functionality, like:

- Location Based Services: see your device(s) on a map: is it stolen or lost? Should it be there? It's even possible to automatically switch to another profile (like blocking internet usage) if the device is out of its workable area.
- Costcontrol: do not only rely on the operators CDR's which report you receive after the damage has been done (like extensive roaming costs during holidays), but allow

alerts to be send when a device's data-transfer or call-logs exceed a set amount. This can be done per day, thus preventing high costs.

- Reporting: use parts of the device's attributes like battery-status or installed apps, to report to CrystalReports kind of modules already present in your company. Why reinvent the wheel?
- Import/Export: most information that goes into the MDM-system can be automated by imports. Enriched data can be exported and used in (external) systems or other use like billing, reporting

Extensions can be developed in-house, but also by third-party companies that keep doing what they're best at, and integrate that with the Mobility as a Service. For this it's essential that these third-parties can interface behind the firewall. Commonly used components are SOAP/Webservices in order to allow an open communication between different parties.

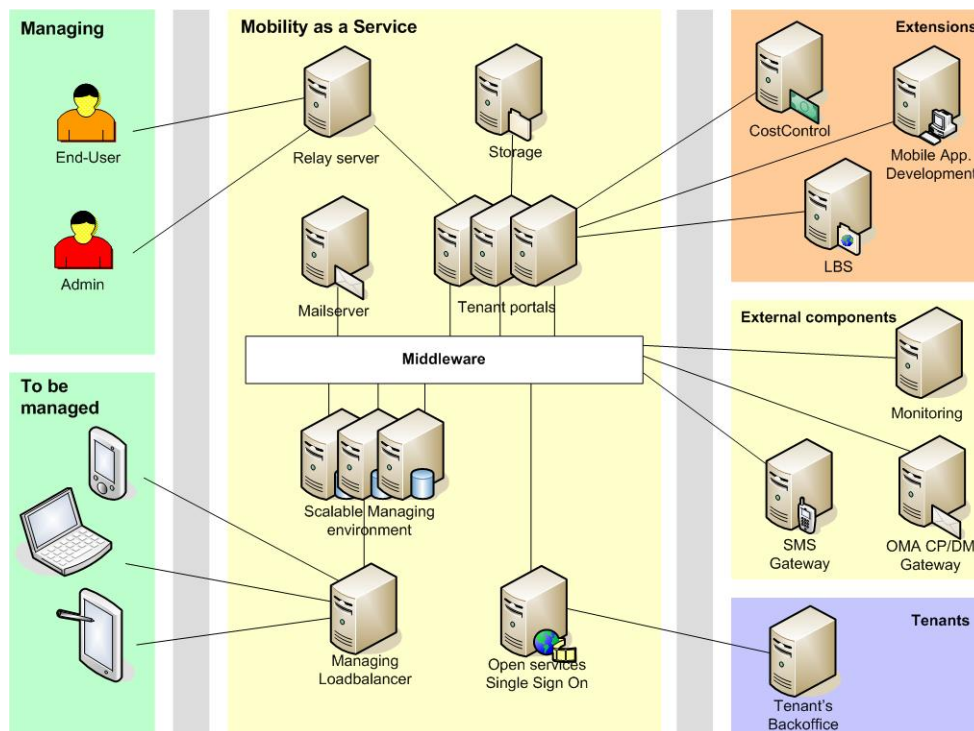
Architecture

Servicing a complete mobile workforce, it is extremely important to choose the right architecture to be able to adapt to high server-loads and be able to support future extensions for more functionality. As Device Management is slowly becoming a commodity, a MaaS should also be open to other external connections passing though firewalls. The components in the MaaS that should be recognized are:

- Managing.
Portal/tenant administrators users who create, configure and maintain their mobile environment, as well as end-users who use the portal for self-help purposes
- To be managed.
The devices (PDA's, smartphone, pencomputers, laptops) that should be managed by the system.
- MaaS environment.
This is the service itself and is accessible in the cloud. It contains components like the tenant's portalservers, the device managing software, storage and a secure & restricted opening into the cloud. The heart of the service is the middleware, which forms the basis into which all component plug into. An email server must be in place to provide communication with admins and end-users. The actual MaaS environment must only focus on the thing it is best at: being a universal platform.
- Extensions.
Allows the service to be of much greater value as it adds proven technology and tools from third-party vendors. In the figure below the online Mobile App Development is an (external) Extension. Extensions could be part of the MaaS itself,

but this would imply to service it in the infrastructure itself, which is not part of the core.

- **External components.**
There are some required external components to be used, in order to run the service. Monitoring the heartbeat of the system should be done from the outside. To provision and instant-message devices, a SMS-gateway and a OMA CP/DM gateway is needed.
- **Tenants-infra.**
Behind-the-firewall systems and databases might be needed to be able to merge a MaaS with an back-office solution at the tenant's location. Examples are existing user-details or in-house websites that need single-sign-on techniques to show parts of MaaS' portal.



Components in a Mobility as a Service

Integration

Extensions are (internal or external) tailor made solutions which integrate with the Mobility as a Service, but they usually only communicate with fixed/internal protocols. In order to be able to allow external back-office systems at the tenant's premises, a restricted opening in the walls of the service should be defined. It allows applications at the tenant's to use data of the MaaS. This can be done using Webservices, where functions like 'give me all

usernames in group A' are possible. It also enables Single Sign-On, which support parts of the MaaS' portal interface to be used by users from the outside, who do not have to login onto the MaaS portal again, as they are already logged in at the tenant's portal. A commonly used protocol for this purpose is SOAP.

Private Cloud

Mobile Services are an interesting opportunity to connect non- cloud systems with other (mobile) applications. Applying the scalability, virtualization, and management paradigm of public cloud computing to large-scale in-house IT infrastructure is basically an evolution of IT management approaches of the past.

Conclusion

A mobile device is a part of the modern mankind, and thus no mobile device is the same. It is called a mobile jungle from an administrative point of view in every business. Delivering Mobility as a Service eliminates the need for knowledge, hard- and software and does not require investments upfront. The engine of the MaaS must be able to service all platforms and must offer a scalable solution to cope with 100.000+ devices.

By offering extra functionalities as Extensions, the MaaS is able to stay focussed on its main goal: device management. Extensions can contain client-specific functions, where the knowhow of this is offered by a third-party partner that is an expert in this area.

Even though the term 'Cloud' seems like a hype, it is still very true that more and more applications will be hosted online. Application that previously interfaced with sources in the same local network now need to go 'outside' to connect to other hosted applications. This will increase the need to interface with service like a MaaS, and thus lowering the need of the MaaS's user-interface. This is even truer when Machine-To-Machine (M2M) techniques will be used.

The strength of a MaaS is servicing the user with ease and keeping the complexity behind the scenes, while external back-end systems can efficiently interact with each other. Only happy users and efficient cloud-services keep the MaaS's eco-system alive.