Enterprise Mobile Service Architecture: Challenges and Approaches

by Longji Tang, FedEx, Wei-Tek Tsai, Vice Chair of AZ-TW Society of Technology, and Jing Dong, University of Texas

Abstract: Today, enterprise systems are integrated across wired and wireless networks. Enterprise Mobile Service Computing (EMSC) is a recent development in distributed computing, and Enterprise Mobile Service Architecture (EMSA) is a new enterprise architectural approach for mobile system integration. This chapter introduces the concepts of EMSC, discusses the opportunities, and addresses mobile constraints and challenges in EMSC. The mobile constraints include aspects relating to mobile hardware, software, networking, and mobility. Many issues such as availability, performance, and security, are encountered due to these constraints. To address these challenges in EMSC, the chapter proposes seven architectural views: Enterprise Mobile Service, Enterprise Mobile Service Consumer, Enterprise Mobile Service Data, Enterprise Mobile Service Process, Enterprise Mobile Service Infrastructure, Enterprise Mobile Service Management, and Enterprise Mobile Service Quality. Each is described with principles, design constraints, and emerging technologies. In order to illustrate a practical implementation of EMSA, the chapter presents a major shipping and delivery services enterprise as a case study to describe the integration of service-driven mobile systems in the enterprise.

Introduction

In a service-oriented enterprise architecture [REF-1][REF-2] there are two major components – services and service consumers. Traditionally, end service consumers access the system mainly through personal computers (PCs), such as desktops and laptops (notebooks) in an end-to-end system. With development of wireless communication and technologies, such as 3G and 4G [REF-3][REF-4], mobile and non-PC devices (tablets and others), such as iPhone, iPad, iPod, and Android-based mobile devices are overtaking PCs as the most widely used communication and Internet access devices. Canalys reported that 488 million smart phones were shipped in 2011, compared to 415 million client PCs that were shipped by vendors in its report “Smart phones overtake client PCs in 2011” [REF-5]. Moreover a recent IDC report revealed that PCs will slip in market share from 35.9% in 2011 to 25.1% in 2016, as Android-based devices will grow from 29.4% share in 2011 to become a market leader in 2016 with 31.1% share. Devices running iOS will grow from 14.6% to 17.3% market share by 2016 [REF-6].

Mobile devices are becoming the major interface for consuming services that are not limited to general phone services, but also include email, internet, entertainment, and social media services. Recently, use of enterprise services, such as mobile search, mobile ecommerce (M-commerce) [REF-3] - shopping and shipping, and mobile payment, from mobile devices are growing rapidly.

Enterprise mobile computing began from first-generation smart phones such as IBM Simon Personal Communicator, the first smartphone released in 1993 (Esposito, 2012). The IBM Simon had a simple operating system called DatalightROM-DOS, 16 MHz CPU, 1 MB RAM and 1 MB storage. Its features included sending and receiving facsimiles, e-mails and cellular pages. It also included many applications including an address book, calendar, appointment scheduler, calculator, world time clock, electronic note pad, handwritten annotations and standard and predictive touchscreen keyboards. Around the same time, there were significant developments in Europe and Asia too. For example, Nokia N9000 Communicator came to the market in 1996 and offered pretty much the features that IBM Simon offered but had a mechanical keyboard instead of the
touchscreen keyboard. Many generations of communicators followed. The first “Symbian” smart phone was Nokia N9210 in 2001. These devices were based on the Symbian OS that was a multitasking OS, supported video capturing and viewing, had browsers, navigation software, email clients, etc. Several handsets from Korea and Japan also came to the market that used Symbian OS since 2001. The first generation mobile devices also included Palm Treo 600, and first generation Blackberry, etc.

The modern era of Enterprise mobile service computing started with the release of the Apple iPhone in the summer of 2007 [REF-7]. The iPhone defined a new generation of smart phones. Smart mobile devices, such as Apple iPhone, Google Android devices, Microsoft Windows 7 and Windows 8 smart phones, have a multitasking operating system, a full desktop browser, Wireless LAN (WLAN, also known as Wi-Fi) and 3G/4G connections, a music player, and supported several of the following features (Firtman, 2010): Mobile Web, GPS (Global Positioning System) or A-GPS (Assisted GPS), Digital compass, Video-capable camera, Bluetooth, Touch and gesture support that allowed an user to touch a screen or make a movement to issue commands, 3D video acceleration, and Accelerometer [REF-8].

The features and capabilities of modern smart phones and smart mobile devices (including tablets, iPad, and iPod) provide a portable channel to connect to enterprise services through mobile applications and web interfaces. The mobile channel is regarded as an extension of the Internet and the enterprise network.

Mobile computing has been growing at a staggering rate across the developed nations. The following Table 1 shows the mobile subscription trend. Although not all mobile subscribers perform mobile computing, specifically in developing countries, the dramatic growth of mobile data plan and internet plan subscriptions in developed countries shows that demand of mobile computing is greatly increasing.

<table>
<thead>
<tr>
<th>Global</th>
<th>Developed Nations</th>
<th>Developing Nations</th>
<th>The Americas</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mobile cellular Subscriptions (millions)</strong></td>
<td>5981</td>
<td>1461</td>
<td>4520</td>
</tr>
<tr>
<td>Per 100 people</td>
<td>86.7%</td>
<td>117.8%</td>
<td>78.8%</td>
</tr>
<tr>
<td><strong>Fixed telephone lines (millions)</strong></td>
<td>1159</td>
<td>494</td>
<td>665</td>
</tr>
<tr>
<td>Per 100 people</td>
<td>16.6%</td>
<td>39.8%</td>
<td>11.6</td>
</tr>
<tr>
<td><strong>Active mobile broadband subscriptions (millions)</strong></td>
<td>1186</td>
<td>701</td>
<td>484</td>
</tr>
<tr>
<td>Per 100 people</td>
<td>17.0%</td>
<td>56.5%</td>
<td>8.5%</td>
</tr>
<tr>
<td><strong>Fixed mobile broadband subscriptions (millions)</strong></td>
<td>591</td>
<td>319</td>
<td>272</td>
</tr>
<tr>
<td>Per 100 people</td>
<td>8.5%</td>
<td>25.7%</td>
<td>4.8%</td>
</tr>
</tbody>
</table>

*Table 1 - Global Mobile Subscriptions in 2011 [REF-9].*

Mobile computing is expected to continue its spectacular growth rate over the next five years. Portio Research predicts that mobile subscribers worldwide will reach 6.5 billion by the end of 2012, 6.9 billion by the end of 2013, and 8 billion by the end of 2016 [REF-9].
Erickson forecasts that mobile subscriptions will reach 9 billion in 2017, of which 5 billion will be mobile broadband connections [REF-9]. The trend of continuous mobile growth is changing the enterprise business customer space as shown in Figure 1:

![Figure 1 - Extending Enterprise Customer Space to Mobile User Space.](image)

At the same time, this trend is greatly increasing the demands of mobile applications and mobile services. Each time a mobile ad is displayed it creates what is termed an “impression”. Compared to 2010, total impressions per user on mobile apps in November of 2011 grew 122%, in which shopping and services increased 105% [REF-10]. The widespread mobile consumer adoption creates a new communication and commerce channel for enterprises, such as mobile sales and services (e.g. shopping and shipping), mobile payments and account access, customer self-services, customer incident reporting, notification and recalls, customer relationship management (CRM), and remote asset monitoring and control including real-time security alerts.

To meet this increasing demand for mobile commerce and services, service providers and enterprises are building and improving their mobile infrastructure, applications, and services constantly. The developing trends are not only impacting the business models of the enterprise, but also its service-driven enterprise architecture. Specifically, these are some questions to ponder:

- Can service-oriented enterprise architecture meet the challenges from both mobile service demand and mobile device constraints?
- Can enterprises integrate their existing services and create new mobile services for the mobile world?

New architectural approaches are emerging, such as the Enterprise Mobile Service Computing (EMSC) and Enterprise Mobile Service Architecture (EMSA) to provide solutions to the challenges in the mobile realm.

This chapter starts out by discussing the issues of enterprise mobile computing. It then explores general concepts of EMSC, their standards, such as wireless network standards as well as mobile web standards; protocols, such as HTTP/HTTPS and languages such as HTML5; and architectural styles, such as SOAP-based Web Services and RESTful Web Services. The core of the chapter delineates EMSA as a service-driven architectural approach to building and integrating mobile systems. EMSA includes several architectural views as follows:

- **Enterprise Mobile Services (EMS):** Enterprise mobile services encompass existing enterprise services, Web applications, Cloud services, and specific mobile services. The characteristics of mobile services and
their design principles are discussed.

- **Enterprise Mobile Service Consumers**: The enterprise mobile service consumers include different mobile devices, such as iPhone, iPad and Android-based devices. The interaction patterns between enterprise mobile service consumers and EMS are presented. Mobile client computing (how to consume EMS) is also discussed.

- **Enterprise Mobile Service Process (EMSP)**: The EMSP is a set of composed and managed services executed in mobile platforms and environments, and consumed by mobile devices for completing a complex business process.

- **Enterprise Mobile Service Data (EMSD)**: The mobile context is a set of dynamic data that is applied for enabling context-awareness in EMSA.

- **Enterprise Mobile SOA Infrastructure (EMSI)**: The EMSI is part of Enterprise SOA (Service Oriented Architecture) Infrastructure, which is a glue to connect wireless network to enterprise services and services in the Cloud.

- **Enterprise Mobile SOA Management (EMSM)**: Due to the multitude of mobile devices and mobile platforms, it is critical to have appropriate management practices in place for managing the enterprise mobile service architecture. In this chapter, The EMSM principles are described and discussed.

- **Enterprise Mobile Service Architecture Quality Ontology (EMSQ)**: The quality ontology includes mobile security, mobile performance, mobile availability, mobile scalability and mobile reliability. These aspects will be discussed.

Finally, the chapter presents a case study and discusses the challenges of EMSC and EMSA in both research and practice. The challenges include mobile enterprise integration and real-time service computing.

**Background**

Mobile computing is a distributed computing model that allows mobile devices and their applications to connect/interact with other mobile devices, mobile applications, and servers or services in a wireless communication environment. It is designed for people who travel or work outside the boundaries of their organizations or homes and enables them to communicate, play, and work, by using mobile devices anywhere and anytime. The mobile computing model extends the traditional computing model which requires one to use stationary computing devices that are connected to the internet or the enterprise network through wire. Mobile computing has two major characteristics that differentiate it from other forms of computing [REF-11]:

- **Mobility**: Implies both physical and logical computing entities that can move, in which the physical computing entities include mobile users and mobile devices that change their locations. The logical entities are instances of running mobile applications, sensors, and mobile agents with wireless connection. The user carrying a mobile device can initiate a real-time connection with other mobile devices and systems if they can connect to a wireless network.

- **Broad Reach**: Mobile users can theoretically be reached at any time. As long as users carry an open device with power, they can be reached instantly, provided they don’t block messages or not accept calls for some duration of time.

The above stated characteristics generate five value-added attributes of mobile computing as shown in Table 2:
Table 2 - Value-Added Attributes of Mobile Computing.

<table>
<thead>
<tr>
<th>Value-Added Attributes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ubiquity</td>
<td>Ubiquity refers to the state and capacity of being available at any location and at any given time. A mobile device such as a smartphone or PDA offers ubiquity.</td>
</tr>
<tr>
<td>Instant connectivity</td>
<td>Mobile devices enable users to connect easily and quickly to the intranet, Internet, other mobile devices, services, and database anywhere and anytime.</td>
</tr>
<tr>
<td>Convenience</td>
<td>It is convenient for users to operate in the mobile environment. All they need is an Internet enabled mobile device such as a smartphone.</td>
</tr>
<tr>
<td>Personalization</td>
<td>Mobile device is owned by individual person with his/her personal choice and usage plan. Mobile devices can be customized for individual consumers.</td>
</tr>
<tr>
<td>Localization of products and services</td>
<td>Knowing the user’s physical location at any moment is the key to offering relevant products and services. Users can easily search and find the products and services nearby (within the realm of security constraints).</td>
</tr>
</tbody>
</table>

Enterprise mobile computing is a new kind of mobile computing model which is designed for enterprises extending their business, infrastructure, and services to the mobile space – mobile users and wireless network. The characteristics of mobile computing and their value-added attributes in Table 2 make enterprise mobile computing an important business and IT strategy.

Specifically, enterprise mobile computing is listed as number one and number two in 2012 top strategy technologies by Gartner’s report [REF-12] in which the “Media Tablets and Beyond” is listed as number one and the “Mobile-Centric Applications and Interfaces” is listed as number two. The fast expanding technology of cellular communications, wireless LAN, and satellite services has been making it possible for mobile users to access enterprise information anywhere and anytime. Figure 2 illustrates the evolution of mobile computing [REF-13].
General Constraints of Mobile Computing

Mobile computing has its constraints and concerns for enterprises. This section describes general constraints of modern mobile computing. Deepak et al., described several mobility constraints and issues of first generation mobile computing [REF-14]. A. K. Gupta characterized the constraints of modern mobile computing as three aspects based on performance impact [REF-15]. The three aspects are:

- Mobile device constraints
- Network constraints
- Mobility constraints

Despite mobile devices getting more powerful since iPhone released in 2007, and modern mobile connection speeds becoming comparable to fixed-line broadband, there are still constraints to be dealt with in the mobile space.

Mobile Device Constraints

The constraints mainly include hardware constraints, software constraints, and communications constraints:
Hardware constraints: The characteristics of mobile computing hardware are defined by the size and form factor, weight, microprocessors, primary storage, secondary storage, screen size and type, means of input, means of output, battery life, communications capabilities, expandability and durability of the device. Table 3 lists hardware data of today’s typical smartphones - both iPhone 4GS and Android smartphone:

<table>
<thead>
<tr>
<th>Device Name</th>
<th>iPhone 4G2</th>
<th>Nexus One</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (in) Weight</td>
<td>4.54x2.31x0.37 (in)</td>
<td>4.7 x 2.35 x 0.45 (in)</td>
</tr>
<tr>
<td>Display</td>
<td>3.5 inches (89 mm) diagonal 1.5:1 aspect ratio widescreen LED backlit IPS TFT LCD 640 x960 resolution at 326 ppi</td>
<td>3.7 in (94 mm) AMOLED with resolution of WVGA (480x800 pixels)</td>
</tr>
<tr>
<td>Input</td>
<td>4 buttons switch microphones, touch-screen, acceleration, orientation</td>
<td>touch-screen Micro-USB; 3.5mm</td>
</tr>
<tr>
<td>CPU</td>
<td>900 MHz dual-core ARM Cortex-A9</td>
<td>1GHz, Qualcomm QSD8250 Snapdragon</td>
</tr>
<tr>
<td>GPU</td>
<td>PowerVR SGX543MP2</td>
<td>chipset, Adreno 200 GPU</td>
</tr>
<tr>
<td>Capacity</td>
<td>Memory 512 MB Storage 16/32/64 GB</td>
<td>512 MB RAM, microSD slot (supported up to 32 GB)</td>
</tr>
<tr>
<td>Camera</td>
<td>8 MP back-side illuminated sensor HD video (1080p) at 24/30 frame/s IR filter</td>
<td>3.7V, 5.3 Wh (1,430 mAh)</td>
</tr>
<tr>
<td>Battery</td>
<td>Rear: 5Mpx, autofocus, LED flash, Geo-tagging, video recording Front:No</td>
<td>1400 mAh Internal Rechargeable Li-ion User replaceable</td>
</tr>
</tbody>
</table>

Table 3 - Hardware of iPhone 4GS and Android Smartphone.

From Table 3, one can see that the limited size and means of input/output are big constraints for designing mobile application interfaces and browsers. Mobile device’s limited capacity also impacts running large applications and storing large amounts of data. Moreover their short battery life impacts system stability, service availability and performance. Compared with PCs, camera and sensors are advantages in most mobile devices. They enable enterprises to create new applications and services using mobile camera and sensor technology.

Software constraints: Mobile devices utilize a broad variety of system and application software. The primary system software is mobile operating system (MOS). The MOS constraints greatly impact mobile application software design and capacity for consuming enterprise mobile services. Popular MOS examples are iOS 5.x, Android OS 4.x, Windows 7.x, Palm OS 1.x. MOS impacts application design through application programming interface (API) for applications and services. MOS also impacts architecture and deployment of mobile applications and services.

Communication constraints: Mobile device can communicate with other mobile devices through wireless networking and send short messages by SMS service. The ability of a mobile device to communicate with
wired information systems and services, such as enterprise applications and services, and Cloud-based services is significant for mobile computing. The communication capabilities of mobile devices directly impact mobile application design.

<table>
<thead>
<tr>
<th>Mobile Device</th>
<th>Network</th>
<th>Connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPhone 4GS (2011)</td>
<td>2G - GSM 850 / 900 / 1800 / 1900</td>
<td>Wi-Fi (802.11) (2.4 GHz only Bluetooth 4.00)</td>
</tr>
<tr>
<td></td>
<td>and CDMA 800 / 1900</td>
<td><strong>Combined GSM/CDMA antenna:</strong> quad-band GSM/GPRS/EDGE (800 850 900 1,800 1,900 MHz)</td>
</tr>
<tr>
<td></td>
<td>3G - HSDP 850 / 900 / 1900 / 2100</td>
<td>Dual-band CDMA/EV-DO Rev. A (800 1,900 MHz) GLONASS, GPS</td>
</tr>
<tr>
<td></td>
<td>and CDMA 2000 1xEV-DO</td>
<td></td>
</tr>
<tr>
<td>Nexus One (2010)</td>
<td>GSM 850 / 900 / 1800 / 1900, 3G HSDPA 900 /</td>
<td>Wi-Fi 802.11 a/b/g, Bluetooth v2.1 with A2DP, A-GPS</td>
</tr>
<tr>
<td></td>
<td>1700 / 2100</td>
<td></td>
</tr>
<tr>
<td>Nokia 1100 (2003)</td>
<td>GSM (2G)</td>
<td></td>
</tr>
</tbody>
</table>

*Table 4 - Network and Connectivity of Mobile Devices.*

From Table 4, one can see that iPhone 4GS and Nexus One can communicate using several different wireless communication standards, unlike older smartphones such as Nokia 1100.

**Mobile Network Constraints**

Mobile network constraints are focused on the connection behavior between mobile devices and mobile network providers in a wireless environment. The type and availability of the communication medium significantly impacts the type of mobile computing application that can be created. Although wireless networking has greatly improved, such as from 2G to 3G [REF-3], 4G [REF-16] and Wi-Fi (WLAN IEEE 802.11) [REF-3], the wireless networks (Cellular networks) have some limitations [REF-14], such as packet loss, high latency due to power restriction and available spectrum, and mobility constraints, specifically in boundary of different networks, weak signal coverage area or strong signal interference area. Advanced mobile broadband (high speed cellular networks), such as 3G and 4G, have different Max Downlink (MDL) for stationary clients and moving clients, for instance 4G MDLstationary client=1Gbps, but MDLmoving client=100Mbps.

**Mobility Constraints**

In general, mobility is a characteristic of mobile objects. In mobile computing, the mobility is a quality attribute of the mobile computing system that includes mobile computation and communication. Figure 3 shows the relationship between mobility and mobile computing. The mobile objects typically move in three spaces – physical, network, and information spaces. The mobility in different spaces has different impacts to mobile computing. The mobility constraints can be defined as adaptability and fault-tolerance of mobile computing systems.
Figure 3 - Relationship of Mobility and Mobile Computing.

**Mobility constraints in physical space (MCPS):** Mobile computing systems should adapt to the mobile behaviors of network disconnection and reconnection, and poor connection as mobile users and devices move in their physical locations (space).

**Mobility constraints in network space (MCNS):** Mobile computing systems should adapt to the mobile behaviors of possible poor performance in a complex network topology space with multiple links and routers when mobile application and service connections are moving in the network space.

**Mobility constraints in information space (MCIS):** Information space consists of the large amounts of data scattered in the information superhighway – the Internet (and Cloud). Mobility in information space means mobile users and applications access the information space when moving in their physical space.

Mobility of mobile software components will negatively impact a mobile service system’s quality of service (QoS) [REF-17], in which QoS mainly includes availability, performance and scalability. The possible temporary disconnection during application code migration process when moving in the physical space, can cause certain components to be unavailable for a short time. The high volume mobile access in peak time, with poor wireless connection, can cause poor system performance. Mobile computing systems should adapt to the mobile behaviors of possible poor performance and scalability when high volume mobile users and mobile applications access the information superhighway – the Internet (Cloud) or access enterprise mobile services in private data centers.
Challenges of Enterprise Mobile Computing

The mobile computing constraints described in the last section are also the constraints of enterprise mobile computing. Additionally, there are also other challenges that are significant to the architecture of enterprise mobile systems as listed below.

Mobile Connection Challenge

Modern enterprise applications and services are built to communicate with networks and the Internet. Users access enterprise applications and services through an enterprise network infrastructure. The network connection is the bridge to enterprise applications and services. The mobile connection between mobile devices and enterprise applications and services often disconnect or are of poor connection quality due to mobility and device portability. The mobile connect behavior causes poor availability and performance of enterprise mobile systems. Therefore one can map the connection issues to reliability, availability, performance, and fault-tolerance challenges for designing enterprise mobile systems.

Mobile Heterogeneous Network Challenge

Unlike most of the stationary computers that stay connected to a single network, mobile devices run into more heterogeneous network connections. While they leave the range of one wireless network transceiver and are routed to another, they may need to change transmission speeds and protocols as well. In some other situations, for instance, where a mobile device in adjacent cells overlap or where it can be plugged in for concurrent wired access, it may have to access several networks simultaneously. Moreover, mobile devices may need to switch interfaces, for example, when switching from cellular coverage to Wi-Fi. The heterogeneous network connectivity makes mobile networking much more complex than traditional networking [REF-3]

Mobile Performance and Scalability Challenge

Because of limited bandwidth, high latency, and mobility of mobile computing, mobile performance and scalability is an important issue. Mobile Internet access is generally slower than direct cable connections, using technologies such as GPRS and EDGE, and more recently 3G as well as 4G networks. These networks are usually available within range of commercial cell phone towers. Higher speed wireless LANs are inexpensive but have limited range (see Table 5). The speed of mobile computing depends on wireless network speed. Table 6 shows max speeds of different wireless networks. The slower networks cause higher latency. A research of mobile Web performance showed that “60% of all mobile users expect websites to load as fast as in a regular browser. But, 75% of all users experience slow load times as their #1 issue.” [REF-18]
<table>
<thead>
<tr>
<th>Wireless Network</th>
<th>Standard Description</th>
<th>Theoretical Max Data Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM</td>
<td>Global System for Mobile communication</td>
<td>12.2 KB/sec</td>
</tr>
<tr>
<td>GPRS</td>
<td>Global Packet Radio Service</td>
<td>60 KB/sec</td>
</tr>
<tr>
<td>EDGE</td>
<td>Enhanced Date rate for GSM Evolution</td>
<td>59.2 KB/sec</td>
</tr>
<tr>
<td>HSCSD</td>
<td>High-Speed Circuit-Switch Data</td>
<td>57.6 KB/sec</td>
</tr>
<tr>
<td>3G</td>
<td>Third Generation wireless network standard</td>
<td></td>
</tr>
<tr>
<td>W-CDMA</td>
<td>Wideband Code Division Multiple Access</td>
<td>14.4 MB/sec</td>
</tr>
<tr>
<td>UMTS</td>
<td>Universal Mobile Telecommunication System</td>
<td>3.6 MB/sec</td>
</tr>
<tr>
<td>UMTS-TDD</td>
<td>UMTS + Time Division Duplexing</td>
<td>16 MB/sec</td>
</tr>
<tr>
<td>TD-CDMA</td>
<td>Time Divide Code Division Multiple Access</td>
<td>16 MB/sec</td>
</tr>
<tr>
<td>HSPA</td>
<td>High-Speed Packet Access</td>
<td>14.4 MB/sec</td>
</tr>
<tr>
<td>HSDPA</td>
<td>High-Speed Downlink Packet Access</td>
<td>14.4 MB/sec</td>
</tr>
<tr>
<td>HSUPA</td>
<td>High-Speed Downlink Packet Access</td>
<td>5.76 MB/sec</td>
</tr>
<tr>
<td>4G</td>
<td>Third Generation wireless network standard</td>
<td></td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
<td>100 MB/sec</td>
</tr>
<tr>
<td>LTE Advanced</td>
<td>Long Term Evolution Advanced</td>
<td>1 GB/sec</td>
</tr>
<tr>
<td>WiMax</td>
<td>IEEE 802.16e</td>
<td>128 MB/sec</td>
</tr>
</tbody>
</table>

Table 5: Wireless Network Speed Comparison [REF-19][REF-20][REF-21]
The above Tables 5 and 6 show maximum or average transfer capacities of single link between an access point and a terminal. However, they do not cover the overall performance of wireless access networks, which is out of scope of this chapter. The mobile network performance is no longer about ensuring that the network is up, it isn’t even about ensuring that capacity is available and coverage is provided as expected. It has become a combination of all three of these elements (availability, capacity, and coverage) in order to provide the network quality of service that customers expect. As smartphone usage proliferates, the devices and applications themselves will also become part of the network performance equation.

**Mobile Security Challenge**

Mobile security issues include mobile device security, wireless infrastructure security, mobile information (data), and information systems (apps and services) security. G.H Forman and J. Zahorjan addressed mobile security risk since 1994 [REF-23]. Amjad discussed mobile and wireless communication security [REF-3]. Dwivedi, et al. described information security issues in mobile computing [REF-24]. Moreover, the stealing and destruction of mobile devices are raising a big mobile security, trust, and privacy issue; since personal identity, email and sensitive data, such as bank account details can be stolen.

Some enterprises are adopting BYOD (Bring Your Own Devices) strategy [REF-25] to improve employee productivity and flexibility. If employee’s devices (iPhone, iPad, etc) with business data and sensitive corporate information are stolen or destroyed, then enterprises will be at risk. To preserve mobile device’s integrity (different devices connect to different service providers) and prevent the cellular network from malicious attacks, the manufacturers of mobile devices and mobile network operators control access to critical mobile components, such as the file system, Internet services, multimedia features, Bluetooth, GPS, and communications, through a rights management system, such as signed keys from a Certificate Authority. However, this is not enough to protect enterprise mobile service systems from malicious attacks through
mobile channels, because mobile networks are more broad, more dynamic and easier to access than their fixed network counterparts. Making enterprise mobile service systems secure is a definite challenge.

Enterprise Mobile Service Architecture

This section presents a new architectural style – Enterprise Mobile Service Architecture (EMSA) as a solution to addressing the challenges of enterprise mobile computing by presenting principles, design guidelines and best practice recommendations. The EMSA is a hybrid architectural style composed of enterprise mobile computing (EMC), enterprise SOA (ESOA), as well as enterprise Cloud service architecture (ECSA) as defined and specified in [REF-26]. The Figure 4 shows the relationship of EMSA and other enterprise architectural styles.

![Figure 4 - Relationship between EMSA and other Architectural Styles.](image)

The EMSA consists of several architectural views as indicated below:

- Enterprise Mobile Service (EMS)
- Enterprise Mobile Service Consumer
- Enterprise Mobile Service Data (EMSD)
- Enterprise Mobile Service Infrastructure (EMSI)
- Enterprise Mobile Service Process (EMSP)
- Enterprise Mobile Service Management (EMSM)
Enterprise Mobile Service Quality Attributes (EMSQ)

The relationship among the architectural views of EMSA is shown in Figure 5:

![Figure 5 - Relationships among Architectural Views of EMSA](image)

Figure 5 is based on the framework proposed in Tang’s PhD Thesis “Modeling and Analyzing Service-Oriented Enterprise Architectural Styles” [REF-27] The EMSA architectural views describe the hybrid enterprise architectural style which includes components and connectors that form an enterprise mobile service system. The figure also depicts the design constraints in terms of the mobile quality attributes and their relationship with the other architectural views.

The EMSA style is an abstraction of a family of concrete enterprise architectures that can provide design
principles and guidelines for designing concrete enterprise mobile service architecture. By Taylor’s definition of architectural style, “An architectural style is a named collection of architectural design decisions that are applicable in a given development context, constrain architectural design decisions that are specific to a particular system within that context, and elicit beneficial qualities in each resulting system” [REF-28]. The EMSA style defined in this chapter can be applied to any enterprise mobile service development context as a high level architectural pattern. The EMSA style will be used for evaluating FedEx enterprise mobile service architecture in the case study section. The rest of this section describes all the architectural views of EMSA and provides guidelines and best practices.

Enterprise Mobile Service (EMS)

The manufacturers of mobile devices often provide built-in services, such as native SMS service, Email service, mobile Web Services, and Cloud services. For example, iPhone 4S provides online services App Stores, such as, iTunes store, iBook store and iCloud. EMS is defined as the service in Enterprise SOA Systems that can be accessed from mobile devices for enabling enterprise m-Businesses (mobile businesses) [REF-29], such as m-Commerce (m-B2C and m-B2B), m-CRM, m-ERP, Wireless Advertising [REF-30] etc.

Figure 6 - Enterprise Mobile Services.

Figure 6 indicates three kinds of services exposed to mobile devices:

- Existing enterprise services
- Mobile specific services
- Cloud services
These services can be categorized based on different architectural styles as shown in Table 7:

<table>
<thead>
<tr>
<th>Mobile Services</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component-based services [REF-27]</td>
<td>The service is built based on component architectural style such as J2EE, .NET, etc. Communication is based on RPC (such as RMI) and XML over HTTP.</td>
</tr>
<tr>
<td>SOAP-based Web services [REF-27]</td>
<td>The service is built based on WS-* architectural style (EWS-*) and uses SOAP as its communication protocol and conforms to Web Service standards.</td>
</tr>
<tr>
<td>RESTful Web Services [REF-27]</td>
<td>The service is built on REST-based Web-oriented architectural style REST [REF-31] and EWOA (Enterprise Web Oriented Architecture). HTTP/HTTPS are its communication protocols.</td>
</tr>
<tr>
<td>Cloud Services [REF-27]</td>
<td>Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)</td>
</tr>
</tbody>
</table>

Table 7 - Mobile Services with Different Architectural Styles.

Hirsch, et al. described the standards, design principles and applications of SOAP-based Web services in [REF-32] It is found that RESTful Web Services are more suitable in many areas of enterprise mobile service computing.

<table>
<thead>
<tr>
<th>Constraints of Mobile Computing</th>
<th>Advantage of RESTful Web services</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limited device capacity of hardware and software.</td>
<td>Architecture simplicity: Client-server model with unified service interfaces.</td>
</tr>
<tr>
<td>Limited bandwidth and battery power.</td>
<td>SOAP-based Web services use XML-based messages that need higher bandwidth and consume more power. RESTful JSON-based messages needs less bandwidth and consumes less power for processing.</td>
</tr>
<tr>
<td>Multitude of devices and Rich content in small mobile devices.</td>
<td>SOAP-based Web services support XML data format only, RESTful Web services support multiple media types: HTML, plain text, XML, JSON, PDF, Atom and more.</td>
</tr>
</tbody>
</table>

Table 8 - RESTful Services as a Solution to Challenges of Mobile Computing.

Table 8 maps constraints in mobile computing to a solution with RESTful Web Services. It illustrates that RESTful Web service is a better alternative for architecture choice for many mobile computing systems. However it does not mean that SOAP-based Web service will go away. Because of its message-level security feature and well-defined message contract, it will still be suitable for many mobile enterprise applications. However, SOAP-based Web service needs to continue improving its performance and reducing bandwidth.

A solution architecture with RESTful Web Service and mobile Cloud service is shown in Figure 7, in which the
Cloud can be a private Cloud or any public Cloud such as Amazon Cloud, Google Cloud or Microsoft Cloud [REF-2, 8, 9, 10].

Adopting Cloud services for enterprise mobile service computing is a new trend which is called Mobile Cloud Computing [REF-11,12]. The Cloud can help enterprises to meet some of the challenges of mobile service computing. The IaaS can help enterprises to build highly dynamic, scalable, and highly available mobile infrastructure services. The PaaS can help enterprises to develop, deploy, and run mobile applications and services quickly and in a cost-effective manner. The SaaS can provide additional mobile service capacities.

Enterprise Mobile Service Consumer

Mobile devices are provisioned with applications that can access and consume services locally or remotely. Enterprise mobile service consumer can be defined as any mobile application or user interface that is provisioned in the mobile devices, and that can access and consume enterprise mobile services for enabling enterprise m-Business. This chapter emphasizes on mobile Web applications [REF-13, 14], and interfacing of mobile devices to enterprise SOA systems with Web technology, such as Web Services, Ajax, Javascript (jQuery mobile), and HTML5 [REF-39].

Figure 7 - Solution Architecture with RESTful Mobile Service or Mobile Cloud Service.
When stationary devices, such as a PC or wireless laptop accesses enterprise web services (SOAP-based or REST-based) through wired connection or WEB access point network, the request is directly routed to the Internet. However, with mobile devices, such as iPhone or Blackberry, when accessing enterprise web services through the Internet, the request/response has to be routed through the wireless operator’s cellular network and wireless operator’s proxy servers. Therefore mobile devices exhibit greater latency when compared to stationary devices for sending message requests to or getting responses from the enterprises services.

Figures 8 and 9 illustrate the typical Web service styles used for mobile computing. SOAP uses the same
URL for all interactions. The SOAP server parses the SOAP message to determine which method to invoke. All SOAP messages are sent with HTTP POST requests. However, REST uses different URLs to address resources. The Web server dispatches a request to a handler (URL addresses a handler) and REST maps the request to standard HTTP methods.

RESTful mobile service interaction is simpler than SOAP-based Web services. Therefore mobile services lean towards using RESTful Web Services. Mobile applications are becoming an important consumer for enterprise services, however, they face many challenges with respect to device variations and constraints. HTML5 as the next generation hypertext language that brings new solutions to some of these issues. HTML5 enables mobile users to experience richer Web applications and improved usability. HTML5 compliant Mobile browsers reclaim the rendering of rich Web content from third-party plug-ins.

HTML5 shown in Figure 10, provides advanced features to build sophisticated mobile Web sites, and helps resolve challenging issues of EMSA as shown in Table 9:
**Constraints & Issues of Mobile Computing**

<table>
<thead>
<tr>
<th>Constraints &amp; Issues of Mobile Computing</th>
<th>HTML5 features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limited capacity of hardware and software.</td>
<td>Rich features for supporting modern mobile Web applications.</td>
</tr>
<tr>
<td>Limited bandwidth and battery power. Issues on mobility, such as lost connection, out of coverage areas.</td>
<td>HTML5 documents use a manifest to list all dependent external resources (i.e. CSS files, JavaScript libraries, etc.). Mobile browsers can use the manifest to cache an entire Web application for offline use, allowing mobile users to interact with a Web app while roaming in and out of coverage areas.</td>
</tr>
<tr>
<td>Issues with variable devices and rich content in small mobile devices.</td>
<td>The Web Storage API allows documents to persistently store data in a mobile browser. Mobile browsers can write data in one browsing session and read it in the next session.</td>
</tr>
<tr>
<td>Issues in performance and scalability.</td>
<td>HTML5 processing speeds are improving, with the notable exception of CSS 3D transforms. Has AJAX support – asynchronous communication. Has Websocket support – this reduces HTML message overhead and provides duplex communication and lower latency. Supports real-time Web applications.</td>
</tr>
</tbody>
</table>

*Table 9 - HTML5 Features for Mobile Computing.*

**Enterprise Mobile Service Data (EMSD) and Mobile Context**

Tang described general SOA data entity [REF-1, 2], that includes various service descriptions, and infrastructure configuration data. SOA Data Entity is also applicable to the Data part of EMSD. However, mobile computing is different from traditional distributed computing. In desktop computing, most parameters in the environment are relatively stable, and has constant context. Likewise, notebook/laptop computers are designed for stationary use so there is usually little variation in the situation surrounding their usage. In contrast, the environment of ultra-mobile computing is characterized by change. The mobile context is a dynamic data set.

In IoT systems (Internet of Things) [REF-15, 16], mobile users will be able to dynamically discover and interact in an impromptu manner with heterogeneous computing environments and physical resources encountered during their roam [REF-36]. Mobile context can be used for enabling this kind of context-aware mobile service systems. Several authors [REF-17, 18] define mobile context concept and discuss its application.

The mobile context can be defined as a set of dynamic data that is applied for enabling context-awareness in EMSA style systems. The context-awareness is defined as a service capacity with full awareness of current service execution environment that includes location, time, and user information. The Table 10 describes four types of mobile context [REF-38] in EMSA. The mobile context can be applied in mobile systems to provide context awareness for applications. Enterprise mobile service providers, such as Google and Facebook, make their services dynamically adaptable to the environment of mobile devices and mobile Web clients by using mobile context-awareness.
Physical context and time context are applied for monitoring mobile service systems, such as detecting error, traffic jam monitoring for providing vehicle routing information, etc. User context awareness provides automatic recommendation and situation-based adaptation of services and products.

<table>
<thead>
<tr>
<th>Mobile Context</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical context</td>
<td>Real physical environment which is accessible by using sensors and other resources deployed in the environment.</td>
<td>User Location, Traffic condition</td>
</tr>
<tr>
<td>Time context</td>
<td>Real time of any activity executing in the mobile system.</td>
<td>Time: minutes, seconds, day, week, month and year</td>
</tr>
<tr>
<td>Computing context</td>
<td>Capacity and resources of mobile computing that can be detected by the mobile system.</td>
<td>Device capacity, Wireless type</td>
</tr>
<tr>
<td>User context</td>
<td>User information which the mobile system can be aware of.</td>
<td>User profile, People nearby, Current social situation</td>
</tr>
</tbody>
</table>

Table 10 - Mobile Context and Context Awareness.

Enterprise Mobile Service Process (EMSP)

Tang et al. specified and discussed the general enterprise service process (ESP) in ESOA style [REF-2, 10]. The ESP is a set of composed and managed services for completing a complex business process, such as online shopping or shipping workflow, or any business transaction process. The EMSP is an extension of ESP in enterprise mobile integration. The EMSP can be defined as a set of composed and managed services executed in mobile platforms and environments, and consumed by mobile devices for completing a complex business process. Traditionally, humans participate in an ESP, by logging in and providing information so that workflow execution can proceed, such as in insurance policy claim or inventory control. In the mobile realm, user interaction screen is delivered directly to the mobile user’s device. Mobile service process can enable real-time, contact-aware and adaptive workflow. In this chapter, the design principles, architecture, and application of the following three types of enterprise mobile service processes are discussed:

1. Mobilizing existing ESP consumed by stationary computers, such as PCs and workstations. This type of EMSP includes m-B2C, m-B2B, m-B2G (mobile business to government), m-G2B [REF-29] which are extensions of B2C, B2B, B2G, and G2B. Figure 11 is a high level view of mobilized ESP. The basic steps for mobile-enabling existing ESP are:
   - Adding mobile devices as consumers of existing ESP
   - Adding mobile access point in enterprise service infrastructure
   - Building mobile apps in devices for enabling to run end user work processes, such as JQuery mobile or HTML5-based mobile Web apps
Exposing existing enterprise service processes to mobile user’s App space through a mobile-enabled interface gateway or broker, such as WSI which is based on SOAP Web service. Recently, RESTful Web services are being adopted for improving performance. Enterprise Web Oriented Architecture style process is extended to EMSP.

Enabling an existing workflow engine in application servers of enterprise data centers to execute workflow within a mobile communication environment

Providing EMSM for monitoring and managing life cycle of mobile workflow

Addressing performance, availability, and security

---

2. Provisioning mobile-based EMSP processes for completing a business task in mobile computing environment, such as in mobile Web Mashups, mobile social service processes, and context-based and context aware workflow (Hackmann, Haitjema, Gill, & Roman, 2006). The basic steps for implementing mobile-based EMSP are:

- Providing a light workflow engine in the mobile device, such as “Sliver” [REF-39] which is a BPEL workflow process execution engine for mobile devices that can execute business processes
- Using visual representations of workflow
- Integrating activity execution with mobile core platform features, such as iOS and objective-C or Android OS and java mobile platform
- Standardizing mobile interfaces and definitions to existing activities
- Enabling service discovery mechanism and service connection for each activity
- Addressing issues relating to mobility, adaptability, performance, availability, and security

There are two types of mobile specific service processes that take advantage of mobility – mobile context, adaptability, real-time and sensor capacity. They are:

- Context-aware mobile service workflow
- Agent-based mobile service workflow

One of benefits of context-awareness for mobile workflow is to increase the adaptability of mobile workflow, so that mobile workflow can be successfully and more effectively executed in a dynamic and mobile environment. The mobile service composition for context-aware cooperation [REF-37] is a good example. Figure 12 shows an initiator who is not capable of executing an ad-hoc process requiring several mobile and stationary services. The context-awareness helps the process to find other devices or services in its vicinity for remote access if local application is not available. The Cisco and GE location-aware mobile healthcare workflow solution is another good industry example [REF-40]. The solution is already benefiting from real-time data on the location and movement of mobile equipment and patients through the GE AgileTrac RFID network to make smart decisions to improve care delivery and hospital operations.

![Figure 12 - Context-Aware Cooperation for Mobile Process Composition.](image)

3. Using mobile Cloud service processes consisting of a set of Web services in the public Cloud and a set of Web services in the private Cloud.

Majority of services in the process and workflow engine are located in the enterprise data centers. However, for mobile Cloud service processes, the service process as well as the majority of services and service process engines as well as management systems is located in the Cloud data centers. The architecture principles used in this case are a combination of the principles used in both mobile and Cloud computing paradigms.
Enterprise Mobile Service Infrastructure (EMSI)

The SOA infrastructure is the heart of ESOA [REF-26] and it supports the transformation of business in an enterprise or between enterprises into a set of managed services or repeatable business tasks, that can be accessed over a network when needed. The network can be a local network, the Internet, or a wireless network. The EMSI is an extension of SOA infrastructure defined in ESOA style. One of the challenges of enterprise mobile computing is that the EMSI depends on one or more wireless networks operated by mobile network providers, such as AT&T, Verizon, etc. Any of the requests from any mobile service consumer to an enterprise mobile service (EMS) must go through the operator’s backbone. Therefore the backbone impacts reliability, performance, security, and availability of enterprise mobile services. Choosing the right backbone with solid service level agreements is a very important consideration for enterprise mobile service infrastructure. Figure 13 shows a typical enterprise mobile service infrastructure with two parts:

- Wired SOA infrastructure that is in the enterprise data center
- Wireless infrastructure that is outside of the enterprise
- There are three major types of wireless networks in EMSI including:
  - Cellular wireless networks
  - Wireless LAN network (Wi-Fi)
  - Satellite Networks
- img
- Figure 13 - Enterprise Mobile Service Infrastructure.
- The core of EMSI includes the following components:
  - Existing wired infrastructure
  - Mobile applications and services
  - Mobile operating systems
  - Mobile devices
  - Mobile endpoint security agents
  - Mobile network which includes WLAN (Wi-Fi), 2G/3G/4G wireless networks, and/or satellite network
  - Enterprise network security perimeter, such as firewall (Mobile security is discussed in a later section)
  - Tunnel gateway such as MEAP services
  - Authentication service and user directory such as SSO and LDAP service
  - Application servers
  - Data storage
  - Content repository

As the EMSI is an extension of the SOA infrastructure, the infrastructure ontology defined in [REF-27] shows that EMSI is a connector for all of the components that include the components in EMS, Enterprise Mobile
Service Consumer, and EMSP. The architectural constraints of EMSI are parts of EMSQ.

Mobile service management and mobile service quality are important aspects that impact the delivery of enterprise mobile services. The next two sections deal with these aspects.

**Enterprise Mobile Service Management (EMSM)**

The EMSM is an extension of the SOA Management of ESOA style. As discussed earlier, mobile service computing is different from traditional service computing. Except for following general SOA management principles, the EMSM has its specific aspects and principles based on mobile constraints, specifically on mobility, real-time, and highly dynamic behaviors. For instance, mobile web content is different in many ways from Web-to-desktop content. The mobile world is fragmented, there are hundreds of mobile device models, with different form factors, operating systems, and browsers. On the PC-Web world this is not an issue since both user interfaces and delivery mechanisms has been standardized for years. However, in the mobile world, web content must be routed to the user through an operator network, and additional operator specific content may be added during the download. Also, different devices render content differently, so when the Web server detects a specific device type it may choose to send a variant of the generic content to the requesting device. Finally, different devices are capable of handling the various components of a download differently (for example, images, text, and third-party content), which can dramatically vary download time.

The comparisons of Web content on a mobile device and on a desktop are shown in Table 11:

<table>
<thead>
<tr>
<th>Web Content on a Mobile Device</th>
<th>Web Content on a PC</th>
</tr>
</thead>
<tbody>
<tr>
<td>The mobile device connects to the mobile operator network.</td>
<td>PC connects to the ISP (Internet Service Provider)</td>
</tr>
<tr>
<td>Every mobile request goes through the operator's backbone. The wireless networks can be dynamically changed by mobility. Finally, the request is routed to the Internet for downloading content from web servers.</td>
<td>ISP routes through the Internet to download content from web servers.</td>
</tr>
<tr>
<td>Content server may send different content depending on device capability.</td>
<td>Content server sends same content for every request.</td>
</tr>
</tbody>
</table>

*Table 11 - Difference between Mobile Web Content and PC Web Content.*

Table 11 shows challenges in EMSM, specifically the following issues:

- Mobile Device Management (including BYOD management) [REF-41]
- Mobile Network Management
- Mobile Data and Context Management
- Mobile Service and Apps Management
- Mobile Performance and Security Management
- Mobility Management
Mobile Enterprise Applications Platform (MEAP) is a multi-channel gateway for enterprise mobile application management. A comparison of Mobile Point Solution and MEAP Solution is shown in Figure 14. Obviously, MEAP solution facilitates higher manageability that can provide more operational flexibility and cost control for enterprises (Sybase Whitepaper on Mobility Advantage). MEAP addresses the difficulties of developing mobile software by managing the diversity of devices, networks, and user groups at the time of deployment and throughout the mobile solution’s lifecycle.

![Comparison of Mobile Point Solution and MEAP Solution](image)

**Figure 14 - Comparison of Mobile Point Solution and MEAP Solution.**

Mobile application and service monitoring is one of the most important components of EMSM. Without a monitor in the EMSM, performance and security management is not possible. MEAP is a middleware that manages connectivity and security, and can integrate with mobile monitors. For small and middle size enterprises, Cloud-based mobile service and application management is one of solutions. The Compuware Gomez provides a SaaS Cloud mobile monitoring service that can monitor both mobile websites and native mobile application experiences across platforms such as iOS and Android (Compuware Mobile Monitoring webpage).

**Enterprise Mobile Service Quality Attributes (EMSQ)**

Tang et al. proposed a SOA quality tradeoff ontology that has a set of architectural constraints for designing general enterprise SOA style systems. An ESOA system should meet certain SOA quality attributes to achieve required Service Level Agreement (SLA) and Quality-of-Service (QoS). [REF-42][REF-43][REF-44][REF-45] list the following major SOA quality attributes for general enterprise SOA information systems:

- Performance
- Reliability
- Scalability
- Reusability
Maintainability
Security
Cost
Interoperability
Availability
Flexibility
Manageability
Agility
Simplicity
Consciousness
Accountability

[REF-44] shows that all quality attributes are not equal and there are constraints when trying to meet different attributes, such as the flexibility which may impact performance. The tradeoff of SOA quality attributes is one of the key design principles for designing enterprise SOA systems. The EMSQ is an extension of SOA quality attributes in EMSA. [REF-42][REF-43] describe major quality attributes specifically for mobile service computing and includes adding the following quality attributes to the above list:

Adaptability
Mobility

Mobile Performance: An enterprise mobile service system may consist of different components (web services, web farm, application servers, data storage, and mobile devices) and connectors (wired network, wireless network, Cloud, and sensor network, even satellite network). The figure 15 shows a simple end-to-end mobile service system. In the sample case, the elapsed time for a mobile device to get data from the enterprise web service is about $T_{mobile} = t_1 + t_2 + t_3 + t_4 + t_5 + t_6$ (seconds). On the other hand, for a stationary device PC, the elapsed time will be $T_{PC} = t_0 + t_4 + t_5 + t_6$ (seconds). Since $t_3 \approx t_0$, the performance of the end-to-end mobile service system greatly depends on the performance of the networks in EMSI, specifically on its wireless network performance. Tables 5 and 6 show that different mobile communication standards have different wireless network performance. For instance 4G is not only faster than 3G, but also provides more QoS for mobile systems [REF-46]. Moreover, Wi-Fi standard IEEE 802.11ac is much better than IEEE 802.11a, since the data rate (up to 1300 Mbps) of IEEE 802.11ac is much faster than that (54Mbps) of IEEE 802.11a.

Different wireless service providers support different standards with different cost, therefore properly choosing mobile standards and their operators is the first consideration. From Tables 3 and 4, it can be seen that different mobile devices have different performance. Choosing wireless carrier and selecting supported mobile devices are tied to the cost attributes. Form architecture design point of view, mobile service system design with different approaches also greatly impacts application level mobile performance. For example, adopting RESTful Web service may have better performance than using SOAP-based Web service in mobile SOA system. However its security issues needs to be addressed in RESTful mobile Web service system design. We have also shown that mobility has negative impact on mobile performance in the section on Mobility Constraints.
**Mobile Security:** The mobile application and communication security has been addressed in the book “Engineering Wireless-Based Software Systems” [REF-47]. The InformationWeek reports on the State of mobile security [REF-48] shows that mobile security is a concern for enterprise mobile users. The mobile device is more vulnerable than a wired computer. Therefore, building end-to-end mobile security into a mobile service system is an important consideration in EMSA. Both wired as well as the wireless infrastructure of the enterprise needs to be secured. One should not only consider the traditional security mechanisms, such as authentication and authorization of users, multi-level security check point and access control, but also consider mobile device security, wireless network security, mobile application security, and mobile data as well as content security [REF-49][REF-50]. In general, *Enterprise mobile security* includes the following core functions:

- User authentication (including PIV card)
- Device, mobile app and enterprise services access control
- Stored data encryption and end-to-end encryption for sensitive data, such as credit card
- Application whitelist/blacklist
- Content filtering and malware protection
- Security event monitoring, logging, and response
- Data leak protection & removable storage control

Enterprise mobile users include employees, registered customers, and other users. Enterprise mobile service system should consider defining different policies for access control to enterprise resources based on the different security levels required for each resource. Evidently, employees can be allowed to access more resources with higher security level than registered customers and other users. The registered customers can be allowed to access more resources and app features than other users. For public mobile Cloud services, the registered or subscribed customers can be divided into different subscriber classes, such as Gold, Silver, Copper, etc. depending on their SLA and payment. Gold customers pay more than Silver and Copper customers, and will be allowed to access more resources and features. Other mobile users may only be allowed to access limited resources and features.

**Mobile Adaptability:** Mobile adaptability can be defined as the ability that a mobile service system or mobile application can adapt to the mobile environment where mobile devices are in different mobility spaces such
as physical space, network space, and information data space. The mobile adaptive behavior discussed earlier arises due to a certain mismatch between mobile service or supply, and service consumer demand for resources. The mismatch often happens in low-level system resources, such as bandwidth, memory, CPU or battery power [REF-50][REF-51]. There are many different approaches for achieving adaptability in different circumstances and scenario. Kakousisa et al., investigates software adaptation of mobile and ubiquitous computing in [REF-51]. The general mobile software system adaptation loop is shown in figure 16.

![Figure 16 - System Adaptation Loop in Mobile and ubiquitous Computing.](image)

The Table 12 shows different types of mobile adaptability and corresponding strategies, framework, and architecture.
<table>
<thead>
<tr>
<th>Adaptablety</th>
<th>Description</th>
<th>Strategies, Frameworks, Solution Architecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client and server adaptability</td>
<td>Application-aware adaptation is to use a collaborative partnership between mobile systems (OS) which provides total transparency: apps are completely aware of mobility and each mobile application copes with mobility on its own.</td>
<td>The Odyssey is a solution architecture for application-aware adaptation by multiple applications using diverse data types [REF-12]. However it does not address replication of objects and dynamic reconfiguration.</td>
</tr>
<tr>
<td>Server/Network adaptability</td>
<td>The mobile adaptability is in the server /network level for handling network adaptive behavior, such as available bandwidth changes, disconnection due to user move, or bad battery.</td>
<td>[REF-13] proposed a mechanism to store information, and replicate or cache objects when network disconnection happens.</td>
</tr>
<tr>
<td>Proxy adaptability</td>
<td>The mobile adaptability is in the service proxy.</td>
<td>[REF-14] proposed a TranSend system which is application-transparent and supports dynamic reconfiguration.</td>
</tr>
<tr>
<td>Proxy and client adaptability</td>
<td>The mobile adaptability is in both the proxy and client.</td>
<td>MobiGATE [REF-15] is application-transparent and supports dynamic reconfiguration.                         It applies coordination theory for mobile service composition and configuration</td>
</tr>
<tr>
<td>Middleware adaptability</td>
<td>The mobile adaptability is handled by mobile middleware that supports QoS-aware or policy-based adaptability.</td>
<td>[REF-16] discussed several mobile middleware for mobile adaptability, such as reflective middleware, and policy-based middleware. [REF-17] proposed agent-based middleware. [REF-18] proposed a QoS-Aware mobile middleware.</td>
</tr>
<tr>
<td>Client adaptability</td>
<td>The mobile adaptability is handled by dynamic user interfaces and on-demand-code.</td>
<td>Using on-demand-code, such as Javascript and other technologies, like HTML5, CSS3 [REF-19] to create dynamic user interface for adapting to different types of mobile devices.</td>
</tr>
</tbody>
</table>

Table 12 - Types of Mobile Adaptability.

**Mobility:** This section describes the impacts to other system quality attributes by mobility as well as its impacts to system design. Enterprise architecture for Mobility extends enterprise service and application computing capacity and communication channels from non-mobile paradigm to mobile paradigm, however, the mobility of
software components may negatively impact the QoS of mobile service system [REF-60], such as availability and performance due to the fact that system may be unavailable or may be slow during the migration process. To overcome mobility’s negative aspects, mobile systems should be designed with support for certain adaptability as detailed in the last section and for fault-tolerance as described earlier. There are two types of mobility in mobile computing systems [REF-60]:

- Physical Mobility: is the ability that mobile users with their devices can access information systems unimpeded while they are moving across different physical locations
- Logical/Code Mobility: is the capacity that the modules or components in the mobile system can dynamically move across servers during system execution
- There are two kinds of logical mobility (LM) [REF-61][REF-62].
- Stateful Mobility or Strong Mobility (SM): This is the ability of a system to allow migration of both code and the execution state of an executing unit to a different computing environment.
- Stateless Mobility or Weak Mobility (WM): This is the capacity of a system to allow just code movement across different computing environments.

The Table 13 shows a classification of logical mobility mechanism.
<table>
<thead>
<tr>
<th>Logic Mobility / Mobility Mechanism</th>
<th>Code and Execution State Management</th>
<th>Data Space Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stateful/Strong Mobility</td>
<td>Migration</td>
<td>Binding removal</td>
</tr>
<tr>
<td></td>
<td>■ Proactive</td>
<td>Network reference</td>
</tr>
<tr>
<td></td>
<td>■ Reactive</td>
<td>Re-binding</td>
</tr>
<tr>
<td></td>
<td>Remote cloning</td>
<td>By copy</td>
</tr>
<tr>
<td></td>
<td>■ Proactive</td>
<td>By move</td>
</tr>
<tr>
<td></td>
<td>■ Reactive</td>
<td></td>
</tr>
<tr>
<td>Stateless/Weak Mobility</td>
<td>Code shipping</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Standalone code</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Synchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Asynchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Immediate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Deferred</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Code fragment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Synchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Asynchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Immediate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Deferred</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Code fragment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Synchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Asynchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Immediate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Deferred</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Code fragment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Synchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Asynchronous</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Immediate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ Deferred</td>
<td></td>
</tr>
</tbody>
</table>

*Table 13 - Classification of Logic Mobility Mechanism.*
Compared with non-mobile client-server style system in which the server has know-how logic and resources needed for providing a given service, the distribution of such know-how logic (code) and resources varies across the components in mobile systems. There are three types of mobile code systems for supporting logical mobility [REF-61][REF-62]: remote evaluation, code-on-demand, and mobile agent as shown in Table 14.

Adopting proper architectural style is important when designing mobile service systems. The design guideline for selecting appropriate style is shown in Table 15. Fuggetta, et al., evaluated several code mobility technologies and discussed applications as well as issues of code mobility in [REF-61]. The technology of Logic/Code mobility is a promising solution for the design and implementation of large-scale distributed mobile systems because it overcomes drawbacks of traditional client-server computing. As the execution units belong to different users, and computation environments are managed by different organizations; and communication may take place through an insecure infrastructure (Internet), the security is a concern to use logical/code mobility technology. [REF-50][REF-61][REF-63] provide security solutions to code mobility.

<table>
<thead>
<tr>
<th>Type of System</th>
<th>Architecture Description</th>
<th>High-Level Diagram</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Remote Evaluation (REV)</strong></td>
<td>There are source hosts and destination hosts. A component in source host has know-how logic but not the resources needed for performing the service. The component is transferred from source host to destination host where it is executed using the available resources. The result of the execution is returned to the source host.</td>
<td><img src="image" alt="Figure 17 - Remote Evaluation" />](image) [REF-24]</td>
</tr>
<tr>
<td><strong>Code-on-Demand (COD)</strong></td>
<td>The needed resources are available in a host A which does not have know-how logic. The subsystem in host A requests the component (s) providing the know-how logic from remote host(s), such as host B. The logic will be executed in host A.</td>
<td><img src="image" alt="Figure 18 - Code-on-Demand" /> [REF-25]</td>
</tr>
<tr>
<td><strong>Mobile Agent (MA)</strong></td>
<td>A software component (MA) located on a given host A, which has know-how logic, some of execution state and has access to some of the resources needed to provide a service. The MA with its state and local resources may migrate to destination host that may have remaining resources needed for providing the service.</td>
<td><img src="image" alt="Figure 19 - Mobile Agent" /> [REF-26]</td>
</tr>
</tbody>
</table>

Table 14 - Architecture of Mobile Systems with Logic Mobility Mechanism.
<table>
<thead>
<tr>
<th>Design Style / System</th>
<th>Client-Server</th>
<th>Remote Evaluation</th>
<th>Mobile Agent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non Mobile</td>
<td>Appropriate</td>
<td>Code represented as data. Code receipt and execution must be programmed explicitly.</td>
<td>Code and state represented as data. Execution and state restore must be programmed explicitly.</td>
</tr>
<tr>
<td>Mobile with Weak Mobility</td>
<td>Degenerated code. Unnecessary execution units are created</td>
<td>Appropriate</td>
<td>State represented as data. State restoring must be programmed explicitly.</td>
</tr>
<tr>
<td>Mobile with Strong Mobility</td>
<td>Degenerated code. Unnecessary execution units are created. Unnecessary state migration.</td>
<td>Unnecessary state migration. Unnecessary overhead for migration.</td>
<td>Appropriate</td>
</tr>
</tbody>
</table>

*Table 15 - Appropriate Architectural Styles for Different Systems.*

We covered the major quality attributes in enterprise mobile service architecture. [REF-42] proposes a framework for quality metrics in mobile-wireless information systems. The author builds quality metrics and maps each metric to issues of mobile-wireless computing, such as “Narrow band”, “Connection stability, and “Security and privacy”. Moreover, the author also builds a mapping from mobile quality metrics to standard quality characteristics.

**Case Study: Enterprise Mobile Solution**

FedEx is a successful transportation/shipping company in the world. Its mobile solution (Fedex Mobile) is a noteworthy implementation in enterprise mobile service computing. FedEx mobile solution includes mobile web solution and mobile sensor–based real-time services. The FedEx mobile web solution consists of:

- Mobile App Store which carry mobile applications for different mobile devices with different OS such as iPhone/iPad, Android and Blackberry
- Core services, such as shipping, tracking, rating, scheduling, pickup, and find location
- A solid SOA infrastructure [REF-67] as a foundation of its enterprise mobile service architecture
- Secure communication and data access with its user management and remote control to protect its business and enterprise services

FedEx CIO Rob Carter in his “Dominant Design” presentation [REF-68] in InformationWeek provides an in-depth look at how FedEx is building its data centers which are based on SOA and private/hybrid Cloud enterprise infrastructure architecture. Its core components include virtualized servers, modern network, intelligent storage, and SOA software architecture. The enterprise infrastructure architecture is the foundation of FedEx mobile solution.

Another FedEx mobile solution is to provide sensor-based real-time mobile services. The new product SenseAwareSM powered by FedEx [REF-69] shows that sensors and the Internet of things (IoT) will dominate its mobility strategy in the future. This device is dropped into a package and can relay back location, temperature, data and even radioactivity levels if needed. Internet of Things (IoT) is one of the emerging
technologies [REF-70][REF-71]. Enterprise mobility as well as mobile service computing is basic technology for IoT realization. The sensors connect many different disciplines, including customer service, data management, analytics, and business intelligence.

Unlike traditional mobile devices, wireless sensors are standard measurement tools which can be part of a mobile device or a wireless sensor device, equipped with transmitters to convert signals from process control instruments into a radio transmission. The radio signal is interpreted by a receiver which then converts the wireless signal to a specific desired output, such as an analog current or data analysis via computer software. Wireless sensors are very useful in infrastructure-free environments, such as sensor networks, ad-hoc networks for monitoring physical or environmental conditions such as temperature, sound, pressure, etc., and to cooperatively pass their data through the network to a main location. FedEx SenseAware can be used for customers monitoring shipment in-transit condition in near real-time. If an enterprise has infrastructure-free environments or ad-hoc networks like FedEx, then enterprise wireless sensor service architecture is an important part of the enterprise mobile solution architecture.

As we briefly described, FedEx enterprise mobile solution is built on its solid enterprise SOA mobile architecture which is an instance of EMSA architectural style described in this chapter. As we defined in EMSA, the architectural style consists of seven architectural views. Let us examine the FedEx mobile architecture and see how it maps to a concrete EMSA style architecture.

- FedEx mobile architecture consists of a set of mobile Web services (EMS) – such as FedEx mobile ship, mobile tracking, mobile locator, etc., which support its core business and provide online and offline services to its employees, customers and public users. More information is available from the FedEx mobile portal.
- FedEx mobile architecture provides its mobile applications in Apple store (iOS), Google store (Android OS), and BlackBerry world, for mobile service consumers (Enterprise Mobile Service Consumers).
- FedEx mobile architecture is built on its hybrid private enterprise SOA infrastructure and integrates its existing infrastructure to the enterprise wireless sensor network (EMSI) through its SenseAware mobile innovation product and Rackspace Cloud.
- FedEx mobile architecture supports mobile workflow (EMSP) and allows mobile users to complete shipping process by using mobile devices.
- FedEx mobile architecture is designed with its mobile enhanced SOA management system (EMSM) for managing and securing its data center infrastructure and providing quality of service (QoS) for its mobile users (EMQS).
- FedEx mobile architecture is designed and implemented for meeting quality requirements, such as security, performance, scalability, and availability [REF-72]. FedEx mobile architecture improves its QoS by adopting new technologies and standards, such as x86 virtualized Linux platform, high-speed WLAN, and 3G above Cellular network. Recently, FedEx adopted Bluetooth wireless technology for its data collection device, the FedEx PowerPad, which gives FedEx couriers wireless access to the FedEx network, thereby enhancing and accelerating the package information available to its customers.

The figure 20 describes a high-level view of FedEx mobile service architecture based on public information [REF-72][REF-73][REF-69][REF-71]:

![Image of FedEx mobile service architecture](http://example.com/image.jpg)
Future Research Directions

EMSA is a rapidly developing trend in enterprise IT. EMSA is still in its infancy and is expected to grow substantially through future research work and practices. However, some of the open issues and challenges, such as dynamic resource provisioning, higher availability, performance, and scalability, need further research and development.

Future research trends in mobile computing are pointing in the following directions:

- Extending enterprise mobile service computing to enterprise mobile Cloud service computing (EMCSC) [REF-73][ REF-35]
- Extending EMSA style to enterprise mobile Cloud service architecture (EMCSA) which is a hybrid style of ECSA style [REF-44] and EMSA style.
- Enhancing EMSI capacity and scalability by using Cloud IaaS [REF-73]
- Utilizing Cloud computing principles, such as dynamic resource provisioning, to enterprise mobile service computing design [REF-74]
- Extending mobile service to SaaS for mobile users
- Utilizing Storage as a service for improving mobile storage challenge
- Developing real-time mobile web applications for supporting IoT [REF-70]
- Developing mechanism for mobile QoS-awareness and SLA-awareness [REF-75].
Conclusion

This chapter presented the opportunities, challenges, issues, and solutions of enterprise mobile service computing. A case on FedEx mobile solution illustrates practical enterprise mobile service architecture. The EMSC as a new paradigm of distributed computing and a new architectural style brings great opportunity to enterprise businesses. Mobile applications and services build a new channel between enterprise business and its customers. However, EMSC faces many challenges and issues as discussed earlier. The main challenges include:

- Mobile device hardware and software challenge
- Mobile connection challenge
- Mobile heterogeneous network challenge
- Mobile performance and scalability challenge
- Mobile security challenge

To solve those challenging issues, the chapter proposes an architectural style – EMSA as an architectural solution which provides design guidelines for enterprise mobile application and service system. The EMSA includes a set of design constraints, principles and high-level common structure – components, connectors and infrastructure. The mobile system design constraints include:

- Hardware constraints
- Software constraints
- Communication constraints
- Wireless network constraints
- Mobility constraints
- Enterprise architecture quality constraints, such as performance, security

In summary, the major guiding principles for architecture of enterprise mobile systems are:

- Work around mobile system design constraints
- Meet enterprise software architecture quality and system non-functional requirements
- Tradeoff quality attributes and achieve a balanced whole quality (EMSQ)
- Utilize mobile context in system design, such as designing for context-aware and location-aware
- Apply Logic/Code mobility appropriately
- Choose the right style of mobile services for maximum performance
- Integrate mobile applications with enterprise core services
- Build mobile App stores for supporting multiple mobile devices
- Build enterprise mobile service infrastructure (EMSI) as a foundation
- Build enterprise mobile service management (EMSM) to guarantee end-to-end QoS and SLA
Disclaimer

This chapter appears in Service-Driven Approaches to Architecture and Enterprise Integration, edited by Raja Ramanathan and Kirtana Raja, Copyright 2013, IGI Global, www.igi-global.com. Posted by permission of the publisher.

References


[REF-43] Garofalakis, Stefani, Stefanis, & Xenos, 2007;


Longji Tang serves as a Senior Technical Advisor at FedEx’s Information Technology Division where he has acted as a tech lead and/or architect on several critical eCommerce projects. Currently, Longji is the lead project manager for FedEx.com’s Data Center Modernization project. His research focuses on software architecture and design, service-oriented architecture, service-oriented cloud computing and application, and system modeling and formalism. Prior to his tenure with FedEx, Longji worked from 1995-2000 as an Information System and Software Engineering Consultant at Caterpillar and IBM. He has published more than 20 research papers from numeric analysis to computer applications in Journal of Computational Mathematics, Acta Mathematica Scienia and other publications. After graduating from Hunan University with a Bachelor of Engineering degree in Electrical Engineering in 1980, he worked as an associate research fellow at the Hunan Computing Center from 1980 to 1992. He began graduate studies at Penn State University in 1992 and graduated in 1995 with a Master of Engineering degree in Computer Science & Engineering and a Master of Art degree in Applied Mathematics. Longji has undertaken his PhD studies in Software Engineering as a part-time student at the University of Texas at Dallas since June, 2002. He obtained his PhD degree in 2011.

Contributions

- Modeling and Analyzing Enterprise Cloud Service Architecture - Part I

---

Jing Dong

Jing Dong received the BS degree in computer science from Peking University and the PhD degree in computer science from the University of Waterloo. He has been on the faculty of Computer Science department at the University of Texas at Dallas and consulting in software industry. His research interests include services computing, formal and automated methods for software engineering, software modeling and design, and visualization. He is a senior member of the IEEE and the ACM.

Contributions

- SLA-Aware Enterprise Service Computing - Part I

---


Yajing Zhao

Yajing Zhao received the B.S. degree in computer science from Nankai University in 2005. She received the M.S. degree and Ph.D. degree in software engineering from the University of Texas at Dallas in 2007 and 2010, respectively. She is a software engineer working on real-time multi-threaded systems. Her research interests include software architecture, design patterns, loosely coupled software designs, algorithms and performance, web services, semantic web services, Ontology, cyber physical systems, and real-time systems, network security.

Contributions

■ SLA-Aware Enterprise Service Computing - Part I