Mobile Electronic Memos: a Novel Approach to Mobile Advertising

G. Bartolomeo, N. Blefari Melazzi, R. Glaschick, L. Schnake, and S. Salsano

***Abstract*—This paper describes a new approach to mobile advertising, providing users with a compelling experience through a new versatile aggregator of information called Mobile Electronic Memo. Security feature of this new medium are discussed alike.**

***Index Terms*—Digital Signature, Distributed information systems, Electronic Formats, Messaging.**

I. INTRODUCTION

An ongoing trend for users is to be the very same protagonists of what they consume, instead of simple audience as in former times. The success of social networking, and the rising of new amazing social phenomena driven by technology[1], prove that self-creating, commenting, recommending and sharing contents are perceived as high charming features by end users.

To fit these new users expectations, in next future Internet advertising will take a more personalized form, becoming interactive and open to exploit several different technologies and spreading channels.

This paper introduces a novel approach to mobile advertising, exploiting a new versatile aggregator of information called Mobile Electronic Memo (MEM). A MEM [1] is an electronic note containing a structured set of attributes associated with a specific class of information. In principle, any resource can be described by a MEM: examples include but are not limited to locations, people, services, physical items or Web sites. As it will be described, the intrinsic and almost unique power of MEMs resides in their quality of being, at the same time, an open and versatile aggregator of information, compatible with existing standards, and a cheap and fast context transfer medium, exploitable by existing and future technologies.

II. BOOKING THE SHOW

A Saturday morning, 8:30 AM. Diana is jogging through the green gardens of NY Central Park when an adv for a new comedy in Broadway attracts her attention. Diana has no time to read it thoroughly, but she sees that it displays a visual tag. She points her phone camera to the tag and captures the visual tag through a simple click. While she is resuming her jogging, her phone detects a network connection. It automatically downloads a MEM using the web URL decoded from the captured visual tag.

Few minutes later, Diana decides to have some rest. She sits down and takes a look at the captured MEM stored in her phone's memory. The MEM contains some human readable information: title, dates and time of the show, address of the theatre where the comedy is held, prices, reservation service phone number and web site. As well it contains machine readable information, like a signature from a well known certification authority and the theatre's GPS coordinates. Such information is displayed as icons, thus Diana realizes that the captured MEM contains authentic information, it is not a phishing, and, if passed to a navigator application, it can even drive her to the theatre!

Browsing the MEM, she clicks on "display comments". Her phone browser opens a Web page, containing comments and rates by people who have already seen the show. Some comments are anonymous, other are signed. She realizes that his favourite actor is in the cast and, according to comments, he has given a great performance. Being willing to watch the comedy, she decides to ask his friend John to escort her. She puts a note on the MEM, "would you like to come with me on next Friday?" and, by a click, she sends the annotated MEM to John's mobile as an asynchronous message (similarly to a MMS).

It is midday when lazy John wakes up. Switching on his mobile, he is alerted about the new received MEM. Great! His friend Diana is asking him to go to a Broadway show this evening. Perfect time for a surprise to her. By clicking on the theatre phone number displayed in the MEM he is connected with the booking office. Using his credit card, he buys two tickets. In return he receives a personalized MEM containing a digitally signed electronic ticket for two people, with seat reservation details. John forwards this MEM to Diana, annotating it with the reply "Let's meet at 18:30 in front of your home".

On Friday evening, John and Diana are on the car. Using Bluetooth, John transfers the MEM from his phone to the car navigator which guides them through Broadway streets…

III. FEATURES

By design MEMs are readable both by humans and by computer applications. Users can browse their MEMs for useful information, as they would do with web pages, but in addition they can also start applications which consume MEMs to provide several additional services (e.g. driving

[1] A curious example is given by people appearing in Google Streetview pictures asking Google to not blur their faces to be explicitly recognized by friends and relatives.

users to a theatre, validating a purchased ticket or checking room availability in a nearby hotel).

MEMs can be easily created by service providers or even by users from web pages offering suitable templates for different classes of MEMs (section VI). More interestingly, users can create MEM directly from their mobile phones. Any context information captured by the phone (GPS coordinates, temperature, taken multimedia, etc.) can be automatically inserted into a MEM.

MEMs can be captured from several sources, examples include but not limit to: physical interaction with the environment via Near Field Communication/Bluetooth or Optical capture through a visual code representing the URL for the MEM [2]; location-based capture with GPS coordinates used to query a backend database, returning a number of MEMs describing resources in the nearby; web capture, clicking on web links, explicitly typing an URL in a browser or typing numeric/alphanumeric codes in well known MEM repositories sites.

Once created or captured, MEMs reveal to be a cheap and fast information transfer medium: they may be optionally annotated and transferred as asynchronous messages like traditional SMS and MMS, and they may be shared among users, by making them publicly available through the Web. Public MEMs can be commented, rated and recommended by users, creating social networks around the resources they describe. In order to certify the creator or the sender authenticity, MEMs or part of them can be signed (section V). Even comments and rates from other users contained in a MEM may be signed to prove their authenticity. Finally MEMs can be encrypted for secure transmission or storage purposes.

IV. FORMAT

MEMs can be expressed in XML or JSON [3] (Fig. 1). Conversion rules between these two formats are provided in [1]. Both formats allow straightforward mapping to software objects in an object oriented environment.

Top level tags “META”, “BODY” and “ENCL” are predefined and reserved, together with two additional tags, “SIGN” and “CRYPT”, below explained. Meta-information carried in the “META” tag include an unique serial number assigned to the MEM, a format indicator (defining one or more schema which the “BODY” tag conforms with) information about the creator (an human being or a software process, acting in a way similar to an RSS feed), a creation date and an optional expiration date, a privacy level field (allowing to flag the confidentiality level of a MEM, using levels from public to highly confidential for private communication), information about the sender and other details.

Contents within the MEM are grouped within one single “BODY” tag. Inside this tag, fields can refer to other resources, or even to other MEMs. Such data can be simply referenced through corresponding URLs or can be embedded into the MEM. The reserved field name "ENCL" (“enclosure”) is used to attach an arbitrary content to a MEM. There may be zero to many “ENCL” tags.

```
{ "META":{ "metafield1": "value1",
           "metafield2": "value2"
         },
  "BODY":{ "field1":"value1",
           "field2":"value2"
         },
  "ENCL":[ {...},
           {...}
         ]
}
```

Figure 1. Overall structure of a MEM (JSON notation)

An XPath-like syntax is used to refer to inner tags contained in a MEM. In particular, \META is used to refer to the “META” tag, \BODY is used for the “BODY” tag and \ENCL[i], is used to identify the i-th enclosure.

V. SECURITY AND DIGITAL SIGNATURE

MEMs or part of them may be digitally signed by the creator or the sender, allowing the recipient to verify the validity of the data and the identity of their creator or sender, thus preventing spam and phishing. MEMs can be signed by service providers as well as by end users, in a secure way, directly from their mobile phones using a secret key contained in their USIM [5]. Therefore, MEMs may be used also to implement non-repudiation capability, being suitable for proof of purchases and subscriptions. Encryption and signing can be almost freely combined and applied also partially to a MEM, e.g. it is possible to encrypt the “BODY” tag, but leave the “META” tag in clear text, even if they both are digitally signed.

A digest or a signature is defined for any substructure of a MEM. Digests without signature can be useful if it can be sent via a slow, but trustworthy extra channel, avoiding the handing of certificates and keys. Also, for commercial transactions, a digest may often be sufficient to prove the integrity of MEM data. The reserved tag "SIGN" defines fields for a digest or for a signature, as desired. Fig. 2 illustrates related fields involved in MEM signature.

```
{"SIGN": { "digestAlgo": "sha1",
           "digestEncoding": "Hex",
           "digestData": "3AB...D2",
           "signAlgo": "PGP",
           "signId": "0xC4989402",
           "signEncoding": "Hex",
           "signData": "123456789AB"
}
```

Figure 2. Signature of a MEM (JSON notation)

The “SIGN” tag is placed inside the object to be signed. Its digest is calculated over the fields contained in the object, alphabetically ordered, excluding the “SIGN” tag itself.

Two kinds of signature are defined. The first one builds a digest from the data contained in the “BODY” tag, includes

the digest as a field in the "META" tag, and finally signs the whole "META" tag itself. Signing the whole MEM is an alternative choise, which however indicates a trust into the content of the enclosures as well.

An encrypted MEM is represented by an object with a single tag having the reserved name "CRYPT", which contains the encrypted data, and related meta information: the encoding format ("base96", "base64" or "hex") [6], the encryption method ("DES", "3DES", "AES", "RSA", "X.509", "PGP") and an optional key indicator (a digest on the key, so that the key can be checked before decryption). Fig. 3 illustrates the reserved tag "CRYPT".

```
{CRYPT: {   "data": "...",
            "encoding": "base96",
            "method": "gpg",
            "id": "0xA3B4C5" }
}
```

Figure 3. Encryption of a MEM (JSON notation)

VI. CLASSES OF MEMS

The concept of MEM class defines the minimum set of tags it is expected a given MEM has to contain in relation to the specific kind of information it carries.

It implicitly specifies the minimum set of requirements a MEM enabled application should implement to parse MEMs. At today, the following classes have been identified:

MEMs for Places – this class contains information about public places, private places like houses and flats, shops and activities, generic locations. A MEM for Place can be exported into several legacy standards, like OASIS Customer Information Quality (CIQ) Extensible Address Language xAL [7] and SIP-Presence Information Data Format (PIDF) Location Object (LO) [8]. The basic schema for this MEM can be extended by several types of MEM belonging to this class, e.g. there may be MEMs for airports, banks, hospitals, offices, restaurants, etc. [9].

MEMs for People – MEMs used to describe human beings. This class of MEM is almost completely based on vCard [10] and CPID [11] properties. Also it contains some extra fields common in social networking user profiles. People MEMs are thus easily exportable as vCard and Microformat hCard [12].

MEMs for Events – MEMs representing an activity held at a given time (e.g., football matches, lectures in a classroom, parties, weddings, festivals, etc.). They are modeled according to the iCalendar specifications [13] and can be exported to Microformat hCalendar [14].

MEMs for Physical Entities – MEMs that may be used to model Real World Objects (RWOs). Examples of RWOs include vegetables, animals, vehicles, artworks, etc.

Memo MEMs – MEMs containing information that cannot be classified into the previous four classes. A list of person to be contacted, a "to-do" list and a text or multimedia message, are valid example of Memo MEMs.

Classes are associated to a fully qualified name and defined through schemas. In addition to classes, it is possible for a MEM to specify other custom schemas it might conform with directly in the "format" field contained in the "META" tag (section IV).

VII. CONCLUSION

MEMs are open and versatile aggregators of information, and, at the same time, a cheap and fast context transfer medium, compatible with existing standard technologies and fostering convergence of several different services and applications in an easy and straightforward way.

Despite designed for a broader scope, MEMs seamlessly fit mobile advertising, providing users with a rich and compelling experience. In addition, their provided security features (signature, hashing and encryption) may be usefully adopted to prevent spam and phishing. More information can be found in [1].

REFERENCES

[1] S. Salsano, G. Bartolomeo, R. Glaschick, R. Walker, N. Blefari-Melazzi, "Mobile Electronic Memos (MEMs) for Simple Mobile Services", available at http://netgroup.uniroma2.it/twiki/pub/SMS/TechnicalReports/tr-MEMs.pdf

[2] Broll, G., Siorpaes, S., Rukzio, E., Paolucci, M., Hamard, J., Wagner, M. and Schmidt, A. 2007. Supporting Mobile Service Usage through Physical Mobile Interaction. 5th Annual IEEE International Conference on Pervasive Computing and Communications, White Plains, NY, USA, March 19 - 23, 2007.

[3] D. Crockford, "The application/json Media Type for JavaScript Object Notation (JSON) ", RFC 4627, July 2006

[4] W3C, XML Path Language (XPath), Version 1.0, W3C Recommendation 16 November 1999, http://www.w3.org/TR/xpath

[5] C. Rust, S. Salsano, L. Schnake, "The SIM card as an Enabler for Security, Privacy, and Trust in Mobile Services", ICT-MobileSummit 2008, 10 - 12 June 2008, Stockholm, Sweden, Conference Proceedings, IIMC, 2008, ISBN: 978-1-905824-08-3

[6] J. Linn, "Privacy Enhancement for Internet Electronic Mail: Part I: Message Encryption and Authentication Procedures", IETF RFC 1421, February 1993

[7] OASIS, "extensible Address Language (xAL)" – CIQ V2.0 specification - http://www.oasis-open.org/committees/ciq/ciq.html

[8] M. Thomson, J. Winterbottom "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)" IETF RFC 5139, February 2008

[9] Schulzrinne, et al., RFC 4589, Location Types Registry, July 2006

[10] F. Dawson, T. Howes, "vCard MIME Directory Profile", IETF RFC 2426, September 1998

[11] H. Schulzrinne, "CIPID: Contact Information for the Presence Information Data Format", RFC 4482, July 2006

[12] hCard, http://microformats.org/wiki/hcard

[13] F. Dawson, D. Stenerson, Iinternet Calendaring and Scheduling Core Object Specification (iCalendar)", IETF RFC 2445

[14] hCalendar, http://microformats.org/wiki/hcalendar

[15] The Microformats community, see http://microformats.org/, http://microformats.org/about/, http://microformats.org/wiki