Mobile Codes: Standards and Guidelines
A discussion document

**Publicis Groupe, Hewlett Packard Laboratories,
Gavitec AG – mobile digit, and Neomedia Technologies**

Introduction

The mobile industries must agree on certain matters if 'codes' – 1D and 2D barcodes – are to become the 'mobile hyperlinks' that take users *en masse* to the mobile internet from their camera phones or other camera-equipped handsets (e.g. www.activeprint.org, www.qode.com, reader.kaywa.com). The Mobile Codes Consortium (MC^2, www.mobilecodes.org) is being proposed as a forum for achieving that agreement.

This document gives an overview of areas where the question of agreement arises – without arguing for any particular outcome, since that will be for the members of the MC^2 to decide collectively. In some cases, the MC^2 will decide on firm standards (specifications). Sometimes softer guidelines will be appropriate. In many cases, the industries will agree not to agree. Occam's razor is appropriate here: industries should agree on as little as possible while meeting the goal of enabling a mobile code ecosystem.

Before moving on to codes as launching points into mobile services, consider ordinary hyperlinks in web pages, which are a very successful design point from which to learn. Web hyperlinks have visual information, data and behaviours associated with them. Visual information makes hyperlinks understandable to users: first, users generally know through conventions such as underlining which elements of a web page are hyperlinks and which are not; second, users have an idea of what is at the other end of a hyperlink before they click on it, from the text or image in its anchor, and from the surrounding context. In addition to visible information, there are standards for the data underlying web hyperlinks, principally HTML [1] and URIs ([2]). Finally, different browsers generally behave in ways that users expect when they click on a link, although they may vary by content type and browser configuration to some extent. Usually, there is contextual information in the page suggesting what type of content will be retrieved – for example, another web page, an audio track, or a video. They can always look at the underlying URI for more detailed information.

The fact that some of those aspects are configurable or conventional and others are written down as firm specifications is appropriate. On the one hand, visual aesthetics for hyperlinks are variable, people will always invent new types of content and new applications to handle it, and they will invent better types of browser behaviour in general. On the other hand, browsers have to be written to behave reasonably consistently in some core respects; they should, for example, always retrieve content from the same underlying web resource for a given hyperlink.

Mobile codes – what we need to agree about

Like hyperlinks, codes are also launching points into services and content. Note, however, that the invoked services do not have to be web-based. A code can as easily be used to invoke a service by SMS or voice, for example. In any case, as with web hyperlinks, there are three core areas for agreement about codes as read by mobile phones: *visual and encoding* aspects, centred around the codes themselves; *data* aspects, concerning the syntax and allowed values of the data

encoded within codes; and *behavioural* aspects, concerning the functionality of the code reader software on the user's handset.

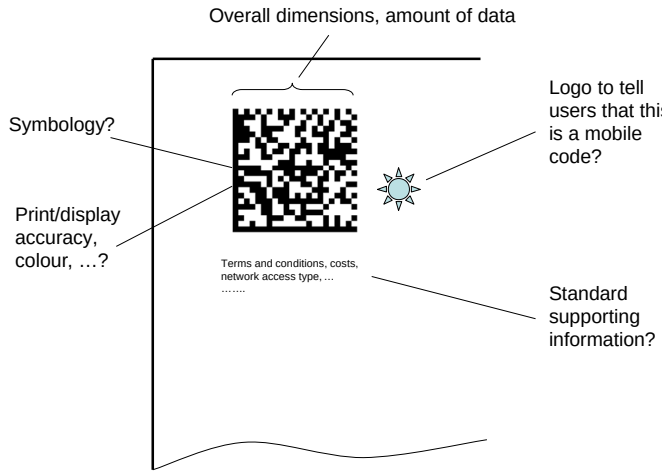

Figure 1. Visual and encoding aspects

Visual and encoding aspects

Figure 1 illustrates aspects of the visual and encoding aspects of codes for which the MC^2 could develop good practice guidelines.

1. Code symbology

Industrial standards exist for barcode symbologies – ways of encoding data into a symbol. QR (Quick Response, ISO/IEC 18004:2000) and Datamatrix (ISO/IEC 16022:2000) are prominent examples in the case of 2D barcodes. In addition, some companies have designed proprietary symbologies, with claimed advantages over more established symbologies. And some companies are using image recognition techniques to link company logos and other symbols to data, using recognition of images or watermarks, and steganographic techniques.

They can't all be the new 'mobile hyperlinks'. It is essential not to confuse users, who need to recognise links to mobile content and services clearly. Equally, users need to know which symbols are *not* mobile hyperlinks – for example, they would quickly become frustrated if they pointed their phones at every company logo they came across, with only a small fraction linked to anything at first.

All types of frustration are to be avoided. Users must be given a good experience that works quickly after little user training, and works in most reasonable circumstances such as lighting conditions, with good feedback to the user.

Experiences of reading codes differ with respect to robustness and aesthetics, and limitations in data capacity. Sometimes those different properties are characteristic of the symbology itself, and sometimes they come about because of the reader – a camera phone or other camera-equipped handset with certain processing and imaging capabilities running a code reader application. The difference is important but largely beyond the scope of this document.

- **Robustness**. Experiences of code reading are not all equal in the following ways:
    - Tolerance of warped codes (for example, on slightly bent newspaper)

- Tolerance of different viewing angles or orientations
- Tolerance of occlusion or damage (for example, partially hidden or scratched codes)
- Tolerance of varying light levels (for example, globally low light levels or local shadowing)
- Decoding speed

Typically the above are influenced by the quantity of data in the code, and the manner in which error correction has been applied.

- **Aesthetics**. Symbologies differ in their look-and-feel. Marketing organizations will not adopt symbols they consider to be ugly.
- **Data capacity and density**. Each symbology is designed to provide up to a certain maximum data capacity (the total number of bits) per code. When that capacity is very limited, codes may have restricted uses. For example, codes with a maximum capacity of a few tens of bits, say, cannot be used to store URLs. But practical data capacities are also to a large extent a function of camera phone optics and resolution. A related concept is the data density of codes: how small on the page can designers make a code to hold a given amount of data? The size of the code has both aesthetic and practical implications. Again, in practice, the achievable density is determined by the reader to a great extent. However, different symbologies in themselves achieve different data densities, due to factors such as the encoding schemes and the size of the registration indicia.

The initial choice of symbology (or a small number of symbologies) for the mobile codes ecosystem should be pragmatic: a code type that just works. It is helpful to bear in mind that evolution will occur. First, camera phone optics and resolution constantly improve, enhancing code reading efficiency and enabling smaller codes. Second, the types of code may evolve with use. Textual hyperlinks in web pages were always designated with underlining at first, but designers developed a variety of more harmonised styles. As long as the rate of change is appropriate, users will accommodate it.

2. Code size and data density

For a code to be readable by a given camera phone, the code must sufficiently in focus, and the image of the code must occupy enough pixels to be resolvable by the recognition software.

Thus, code size needs to be appropriate for viewing distance. For example, codes in a newspaper, a poster at a bus stop and on a billboard will typically be of different sizes since the overall size constraints are different and users hold the phone different distances away.

Equally, the amount of data in the code must be appropriate for camera phones with limited resolutions and optical capabilities, and for practical viewing distances.

Rather than specifying exactly the size of codes and the amount of data in them, it would seem best for the industry to support designers through information and tools. Imagine, for example, a scenario in which a designer proposing to insert a code of a given size with a given amount of data in artwork for a newspaper, would be informed about which phones could read that code, and which could not, based on the manufacturers' own data.

3. Rendering quality

To be readable, codes need to be printed or displayed with sufficient accuracy and colour contrast, on materials with sufficiently low reflectivity. Existing open standards such as QR and Datamatrix codes deal with some but not all of those factors. Moreover, camera phones may be drastically affected in their ability to resolve colour differences by variations in lighting conditions.

4. Meta-information

A barcode itself is opaque to the user: unlike a hyperlink in a web page, there is by default no immediately visible anchor, only whatever information is provided in the context in which the barcode is embedded. That raises the question of whether there is information that must be provided next to the barcode. Examples are:

- terms and conditions of use of the service that the barcode links to
- the tariffs in use
- the type of service invocation (e.g. web vs. voice vs. SMS)

The first two of those already exist for printed instructions to access the mobile internet. However, users might require the type of service invocation to be signposted more clearly than a mention in the fine print nearby.

In addition, a uniform symbol next to the barcode might be needed, to affirm that the barcode links to a mobile service – as opposed, say, to a barcode franked on an envelope; and it might also be used to affirm that the barcode and the associated service meet certain standards of the MC^2.

Not all visual information need by alongside the barcode itself. Active Print's Glass application demonstrates that some meta-information can be in the barcode: the anchor of a mobile hyperlink is in the pop-up text, which the user can observe on reading the barcode, without the need for network access.

Data aspects

Whatever the visual and encoding aspects of a code, its basic job is to contain the hyperlink's data. But what is that data to be, and what is its syntax?

Standards for the data in barcodes exist in Japan, such as NTT DoCoMo's standards [3] for MECARD, MATMSG, MEBKM and LAPL. These respectively allow codes to contain personal contact data, an email message, a web 'bookmark', and invocation of a pre-loaded application.

Web architects have already defined or are developing standards for phone-related content in hyperlinks, including web links, email messages, telephone numbers and SMS messages. These Uniform Resource Identifiers are an alternative, more widely adopted standard. Whatever the choice, the standard must be extensible as new types of service access become possible from the phone.

Finally, codes can contain small user interfaces such as the Active Print project's pop-up text, which appears as soon as a code has been read, enhancing the experience before network access. What mark-up would be appropriate for such interfaces?

Behavioural aspects

Providing good feedback to help the user read codes successfully is one important aspect of code reader behaviour. For example, providing feedback such as pop-up text tells the user clearly when a code has been read. But what if several codes are in the field of view? The user should know which has been read.

Interpretation of the encoded data is the next issue to be considered. How is the code reader to behave when it encounters, for example, a code containing an SMS message? Sending an SMS is not 'normal browser behaviour'. Should the code reader send it directly, or give the user a chance to view or modify it before it is sent? Or should this be configurable by the author?

Other behavioural issues include how the code reader is integrated with the rest of the phone's functionality:

- Should the code reader be distinct from the handset's camera application? If so, should it be distinct from the handset's browser application?
- If it is distinct, should the code reader switch to separate applications, or handle all services and content within its own window space, like the browser typically does?
- To what extent should the code reader be invokable from other applications on the handset?

Many aspects of behaviour such as the above are an opportunity for vendors to distinguish their code readers through innovation and the excellence of their interaction models. But the MC^2 should take a view on whether that should apply in all such respects, for the sake of the user.

What else?

The above discussion indicates the main issues for a mobile codes standards body to consider as falling within or outside its remit. Many readers will be able to think of related areas, or further points of detail. This closing section states some areas that the authors specifically excluded.

The preceding aspects of code reading touch the consumers directly. Also relevant to the consumer experience is the robustness and quality of the services themselves, which they invoke when they 'click on codes'. But it would be too much for any standards body governing mobile codes to deal with those aspects. The W3C does not impose quality standards on the service and content retrieved when a user clicks on a web hyperlink, for good reasons. It is up to all the industries concerned to provide a good mobile internet experience, just as the quality of the PC web experience was raised dramatically in the first ten years of its life.

There are also B2B aspects, concerning interactions between parts of the infrastructure. Standards in this area may eventually fall within the remit of the MC^2 but, again, it is no more necessary to start with than it was for the web. It seems better for now to hope for the day when standards for B2B interactions become an actual issue.

Tim Kindberg, HP Labs, February 2007

[1] HTML 4.01 Specification. http://www.w3.org/TR/html4/
[2] RFC 2396. Uniform Resource Identifiers (URI): Generic Syntax. http://www.ietf.org/rfc/rfc2396.txt
[3] NTT DoCoMo Bar Code Functions. http://www.nttdocomo.co.jp/english/service/imode/make/content/barcode/about/s2.html