モバイルコンピューティング推進コンソーシアム
Mobile Computing Promotion Consortium

MCPC-TR-015

vCard implementation guidelines for Bluetooth profiles

Version 1.00(E)

November 28, 2013

Mobile Computing Promotion Consortium
Technical Committee

Change History

Date	Version	Descriptions
Nov. 28, 2013	1.00(E)	Base version initial release.

Published and copyrights owned by:
Mobile Computing Promotion Consortium (MCPC)
Hasegawa Green Building,
5-12 Shiba-Koen 3-chome, Minato-ku, Tokyo, 105-0011 Japan
TEL： +81-3-5401-1935
FAX: +81-3-5401-1937
EMAIL: office@mcpc-jp.org
WEB SITE: http://www.mcpc-jp.org

How we distinguish binary, decimal and hexadecimal numbers：

- For binary numbers, we append a small letter “b” (e.g. 10b))
- For binary numbers, we insert a space after every four bits. (e.g.：1000 0101 0010b)
- For hexadecimal numbers, we append a small letter “h” (e.g.：FFFFh and 80h)
- All other numbers shall be considered to be written in decimal

Key Words

- “May” means that something is recommended or optional at the free discretion of the vendor.
- “Should” means that although something is not essential, it is strongly recommended. When implementing, the vendor shall take this requirement into consideration and determine whether this is essential or not.
- “Shall” means that something is an essential requirement. For connectivity and specification compliance, the feature must be implemented, and is mandatory.

Trademark Notes

- Company names, product names, service names and the like described in this document are generally trademarked or are registered trademarks of their respective owners.
- In general, the usage of ™ marks, © marks, and ® marks has been omitted in this document.
- In general “ Inc.,” “ Corporation,” “ Co., Ltd.,” have been excluded from company names.

Contributors

Nobuki Asai	ALPINE ELECTRONICS INC.
Shigeru Ota	Clarion Co., Ltd.
Masashi Miura	DENSO CORPORATION
Toshiya Tamura	Fujitsu CORPORATION
Takashi Yamazaki	Harman International
Shigetoshi Hara	Johnson Controls
Hiroyuki Morimoto	Mitsubishi Electric Corporation
Tomoyuki Suzuki	NISSAN MOTOR CO., LTD. for JAMA
Sadamitsu Kanamori	Renesas Solutions Corporation
Masahiko Seki	Sony Corporation
Mitsuyoshi Yasuda	Sony Mobile Communications Japan, Inc.

Table of Contents

1. Introduction

The Phone Book Access Profile (PBAP) and Object Push Profile (OPP) are used for transferring phone book information such as names, phone numbers over Bluetooth wireless technology. The PBAP and OPP are over Object Exchange Profile (OBEX) originated from IrOBEX used for Infra Red communication.

Phone book elements are conveyed in vCard. vCard is widely used for communication systems, not only for Bluetooth and IrDA but also e-mail, WWW, etc. vCard is defined by IMC, and in RFC by IETF and is a universal specification. However there is no definition especially for language dependent information. As a result, there are some unclear points to implementing OPP/PBAP for Japanese usage.

This guideline defines how to specify such language dependent information in vCard and manage interoperability between Bluetooth devices which have PBAP/OPP capability and need to support Japanese language.

Note that MCPC already has published GL-003 Object Exchange Implementation Guidelines. GL-003 describess as using the IrMC compliant Object Exchange function not only for Bluetooth but for general usage cases. Also the TR-010 Phone Book Access Profile Guideline is available. It describes general guidelines for PBAP implementation, not only for usage of Japanese language. This document should be prioritized in case that there are some differences between those documents.

2. Details of recommendation

2.1 Profile requirements

The vCard is used for PBAP and OPP in Bluetooth wireless technology. The usage rules for vCard are a little different between PBAP and OPP.

The following subsection describes specific requirements/recommendations for each profile.

2.1.1 PBAP (Phone Book Access Profile)

Requirements on vCard format are specified in PBAP 3.1.4.

According to the PBAP specification, PSE shall support both vCard 2.1 and 3.0. In addition, UTF-8 shall be used for PBAP even if the phone book in PSE is not encoded by UTF-8.

2.1.2 OPP (Object Push Profile)

Requirements on vCard format are specified in OPP 4.2.1.

According to the OPP specification, phone book application shall support vCard 2.1 even if the application supports other formats. UTF-8 should be used for vCard attribute contents.

2.1.3 Supported profile

vCard data can be transferred by either PBAP or OPP. Supported profiles on the sender depend on each implementation. Therefore it is recommended for the receivers to support both PBAP and OPP.

2.2 Character code

In Bluetooth communications, UTF-8 is normally used as the character code for text data. However phone book data in some mobile phones is stored with Shift JIS and some phones do not convert from Shift JIS to UTF-8 before sending them through Bluetooth. As a result, some may send them with Shift JIS.

As described in Section 22.1.1 and 2.1.2, UTF-8 shall be used in PBAP and should be used in OPP even if phone book data in the server is stored with another character code than UTF-8.

However the receiver is recommended to decode both UTF-8 and Shift JIS to have higher interoperability because some legacy devices may send data with Shift JIS.

The sender should send CHARSET parameters in case of sending Japanese data no matter what character set is used (UTF-8 or Shift JIS). If the receiver does not receive CHARSET parameters and cannot recognize which character set is used, this data should be considered as UTF-8 data.

For the Furigana, Zenkaku (full-width) Hiragana, Zenkaku Katakana and Hankaku (half-width) Katakana can be used for senders. Also receivers should recognize any characters above and be able to sort names properly. Generally Furigana in any kind of characters should not be considered respectively for sorting. For example, “あ” (“a” in Hiragana) and “ア” (“a” in Katakana) have same sounds so that those should be treated as same if those are used as Furigana. This manner is similar to the way for sorting by alphabetical order with capital and lowercase letters.
Note that the details of Japanese Character codes are explained in section 77.

2.3 Encoding

There is no requirement in the specifications on encoding. However it is recommended to use Quoted-Printable for encoding. On the receiver side, Quoted-Printable and 8 bit raw data should be handled because some senders send with 8 bit raw data. This recommendation is applied for all character sets such as Zenkaku Hiragana, Zenkaku Katakana and Hankaku Katakana.

2.4 vCard properties

Recommendations of vCard properties for Japanese language are described as follows.

2.4.1 Property for Furigana

Property for Furigana is normally used for the N (Name) property. It can also be used for other properties such as the FN (Formatted Name) property. However there are not many existing implementations except for the N property currently.

2.4.1.1 vCard 2.1

In case of vCard 2.1, the “SOUND” property should be used to express Furigana. Following to “SOUND” property, the “X-IRMC-“ parameter should be used to specify which property this is used for.

Following is an example of “SOUND” property for the name (“N” property) :

SOUND;X-IRMC-N;CHARSET=UTF-8;ENCODING=QUOTED-PRINTABLE:
=E3=83=A4=E3=83=9E=E3=83=80; =E3=82=BF=E3=83=AD=E3=82=A6;;;

Note that “=E3=83=A4=E3=83=9E=E3=83=80” shows encoded data of “ヤマダ” and “=E3=82=BF=E3=83=AD=E3=82=A6” shows “タロウ” in this example.

2.4.1.2 vCard 3.0

In case of vCard 3.0, the “Sort-String” property should be used to express Furigana.
Additional parameters following the “Sort-String” property is the same as the “SOUND” property in vCard 2.1.

The “SOUND” property can be used even in vCard 3.0. However the usage of the “SOUND” property for 3.0 is different from 2.1. Therefore the “SOUND” property should not be used for 3.0.

Following is an example of the “Sort-String” property for the name (“N” property) :

Sort-String;X-IRMC-N;CHARSET=UTF-8;ENCODING=QUOTED-PRINTABLE:==E3=83=A4=E3=83=9E=E3=83=80; =E3=82=BF=E3=83=AD=E3=82=A6;;;

Note that “=E3=83=A4=E3=83=9E=E3=83=80” shows encoded data of “ヤマダ” and “=E3=82=BF=E3=83=AD=E3=82=A6” shows “タロウ” in this example.

2.5 Others

There are some other implementations than those described in this document. Examples of those implementations are described in Section 6.

3. Terms and Abbreviations

PBAP	Phone Book Access Profile
OPP	Object Push Profile
OBEX	Object Exchange Profile
Sender	The role that sends vCard data. PSE in PBAP and Push Client in OPP.
Receiver	The role that receives vCard data. PCE in PBAP and Push Server in OPP.

4. References

[1] Phone Book Access Profile specification, Bluetooth SIG
[2] Object Push Profile specification, Bluetooth SIG
[3] vCard 2.1, Internet Mail Consortium
[4] RFC2425, ITEF
[5] RFC2426, ITEF
[6] http://en.wikipedia.org/wiki/Furigana
[7] http://en.wikipedia.org/wiki/Japanese_language_and_computers
[8] http://en.wikipedia.org/wiki/Half-width_kana

5. Appendix A (Informative): Explanation of “Furigana”

“Furigana” are Katakana or Hiragana characters used to help reading Kanji characters. Katakana/Hiragana characters express the sound of character (pronunciation) but Kanji characters do not. (Those express the meaning.)

Generally Furigana is used for as a reading aid for children who have not learned Kanji characters so much yet, or for Kanji characters that are difficult to read.

In case of a phone book, Furigana is used for sorting. Sorting with character codes of Kanji is not common in Japanese sorting. In addition, each Kanji character may have several ways of pronunciation. Furigana is also used to clarify the pronunciations.

The most popular sorting method of Japanese is “50 sounds order”. Furigana is useful to make 50 sounds order sorting.

Refer to [6] for more information.

6. Appendix B (Informative): Existing implementations

There are some implementations that are not using the properties recommended by this document. The following shows the examples of existing implementations that are different from the recommendation. If you would like to develop a receiver with higher interoperability to any legacy devices, it may be better to consider supporting the following implementations as well as recommendations in this document.

Some devices use “X-PHONETIC-FIRST-NAME” and “X-PHONETIC-LAST-NAME” as properties of Furigana.
In addition, “X-YOMI-FNAME” and “X-YOMI-LNAME” may be used.

7. Appendix C (Informative): Explanations of Japanese characters

7.1 Characters in Japanese language

Usually Japanese language uses various types of characters as follows:
・Hiragana
Hiragana is used for Japanese native words mainly, and also used with Kanji characters. Hiragana characters express the sounds and do not have meaning themselves.
・Katakana
Katakana is used for words originated from foreign languages mainly. Katakana characters express the sounds and do not have meaning themselves.
・Kanji
Kanji is originated from Chinese “Hanzi” characters. Kanji characters have meaning themselves. Being different from Hanzi in Chinese, many Japanese Kanji characters have a couple of sounds for each character. To ease the reading of Kanji characters, Furigana is widely used for Japanese documents, publications, etc.

7.2 Character codes in Japanese

There are several character code sets for Japanese language. The most popular character code set used in mobile phones is UTF-8. But historically Shift JIS is also widely used in the PC area. Therefore some application may use Shift JIS rather than UTF-8 for the Japanese characters. It may be better to decode Shift JIS as well as UTF-8 to keep higher interoperability.
Note that other character codes for Japanese characters exist such as JIS, EUC, etc. Refer to [7] for more information.

7.3 Zenkaku and Hankaku

In Japanese computing, there are two character styles called “Zenkaku” and “Hankaku”. Zenkaku means the full-width characters and Hankaku means the half-width characters. Any characters including Hiragana, Katakana, and Kanji can be a part of Zenkaku characters but only Katakana can be a part of Hankaku characters. Refer to [8] for more details.